

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

SCUOLA DI INGEGNERIA E ARCHITETTURA
- Sede di Forlì -

*CORSO DI LAUREA IN
INGEGNERIA AEROSPAZIALE
Classe L-9*

Elaborato Finale di Laurea in:
MECCANICA DEL VOLO

**MODELLAZIONE E SIMULAZIONE
DI UN QUADRICOTTERO MULTIROTORE**

RELATORE: *Prof. Fabrizio Giulietti*

CANDIDATO: *Matteo Pantalone*

CORRELATORE: *Ing. Gastone Ferrarese*

Anno Accademico 2014/2015

Sessione I

“When a flight is proceeding incredibly well, something was forgotten”

Robert Livingstone

Indice:

<i>Sommario.....</i>	<i>pag.7</i>
<i>Introduzione.....</i>	<i>pag. 8</i>
<i>Struttura di un Multirotores.....</i>	<i>pag. 9</i>
Cap.1: Fisica di un Quadricottero Multirotores	
<i>1.1 Assetto e Meccanica del Volo.....</i>	<i>pag.13</i>
<i>1.2 Angoli di Eulero.....</i>	<i>pag.16</i>
<i>1.3 Modello Matematico.....</i>	<i>pag.17</i>
<i>1.3.1 Ambiente Matematico.....</i>	<i>pag.17</i>
<i>1.3.2 Equazioni di Eulero-Newton.....</i>	<i>pag.18</i>
<i>1.4 Calcolo dell’Inerzia.....</i>	<i>pag.22</i>
<i>1.5 Forze e Momenti Agenti.....</i>	<i>pag.24</i>
<i>1.6 Dati e Input del Sistema.....</i>	<i>pag.26</i>
<i>1.7 Modello Utilizzato.....</i>	<i>pag.26</i>
Cap.2: Implementazione Modello e Simulazioni	
<i>Matlab e Simulink.....</i>	<i>pag.29</i>
<i>2.1 Implementazione Codice Matlab.....</i>	<i>pag.29</i>
<i>2.2 Implementazione a Blocchi Simulink.....</i>	<i>pag.33</i>
<i>2.3 Manovra di Hovering.....</i>	<i>pag.36</i>
<i>2.3.1 Modello Matlab di Hovering.....</i>	<i>pag.36</i>
<i>2.3.2 Modello Simulink di Hovering.....</i>	<i>pag.38</i>
<i>2.4 Manovra di Salita.....</i>	<i>pag.40</i>
<i>2.4.1 Modello Matlab di Salita.....</i>	<i>pag.40</i>
<i>2.4.2 Modello Simulink di Salita.....</i>	<i>pag.42</i>
<i>2.5 Manovra di Volo Orizzontale.....</i>	<i>pag.44</i>
<i>2.5.1 Modello Matlab di Volo Orizzontale.....</i>	<i>pag.44</i>
<i>2.5.2 Modello Simulink di Volo Orizzontale.....</i>	<i>pag.46</i>
<i>2.6 Confronto e Tempistiche.....</i>	<i>pag.48</i>
Cap.3: Fisica di un Quadricottero con Payload Sospeso	
<i>3.1 Modello matematico.....</i>	<i>pag.49</i>
<i>3.2 Equazioni di Langrange.....</i>	<i>pag.50</i>
<i>3.3 Modello Linearizzato.....</i>	<i>pag.56</i>

Cap.4: Implementazione in Matlab e Simulink

4.1 Implementazione del Modello.....	pag.58
4.2 Manovre Simulate.....	pag.61
4.2.1 Manovra in Hovering.....	pag.62
4.2.2 Manovra in Hovering con Pendolo in Movimento.....	pag.64
4.2.3 Manovra in Rollio.....	pag.68
4.2.4 Manovra in Beccheggio.....	pag.71
4.2.5 Manovra in Imbardata.....	pag.73
Conclusioni e Sviluppi Futuri.....	pag.77
Bibliografia.....	pag.78
Sitografia.....	pag.78
Ringraziamenti.....	pag.79

Sommario:

Nell'elaborato presentato, si è deciso di volgere lo studio alla descrizione e analisi del movimento di piattaforme multirotores tramite modellazione e simulazione computazionale. In particolare si è analizzata l'attitudine di un quadricottero privo di sistema di controllo automatico. Per la scrittura e l'implementazione degli algoritmi risolutivi è stato utilizzato l'ambiente Matlab.

Il codice realizzato permette dopo successiva introduzione di alcune variabili relative a velivolo ed assetto di ottenere l'evoluzione di parametri di volo che descrivono lo stato del drone nel tempo attraverso una progressiva integrazione numerica.

In questo modo è possibile simulare teoricamente qualunque modello di quadricottero conoscendo i suoi parametri costruttivi ottenendo così una modellazione preliminare da lanciare nel simulatore.

Nella tesi viene utilizzato un unico modello realizzato in CAD Solidworks da cui sono stati ricavati i dati necessari.

La tesi si compone di: un'previa trattazione semplificata di fattibilità di manovre semplici di un multirotores di tipo quadricottero con un approccio di Eulero-Newton ed ,in seguito, in presenza di un carico sospeso tramite cavo considerato inestensibile si sono analizzate delle evoluzioni nel tempo con un'approccio questa volta lagrangiano. Infine vengono trattati brevemente osservazioni conclusive e possibili sviluppi di questo lavoro di tesi.

Introduzione:

Il progetto della “vite aerea” di Leonardo Da Vinci nel periodo rinascimentale sarebbe stato solo il progenitore di quella che è stata una evoluzione tecnologica senza sosta sino ad oggi dettata dal più grande dei desideri dell’uomo: volare.

Ad oggi non solo si può realizzare questa, oramai comune, azione ma è altresì possibile avere il controllo su un qualsivoglia velivolo restando comodamente seduti.

Il protagonista che incarna questa nuova concezione del volo è il drone.



Fig.1: Modello di un drone prodotto dalla DJI, l' F550.

Come spesso accade nella storia dell’aviazione, la tecnica e la qualità dei velivoli si sono sviluppate dapprima in ambito bellico durante le due guerre mondiali e lo stesso vale per i droni.

La tecnologia sempre più spinta permise poi nella guerra fredda di ridurne progressivamente le dimensioni fino a quelle che noi conosciamo.

Vista la crescente commercializzazione e la maggior fruibilità del mezzo in seguito l’ENAC (Ente Nazionale Aviazione Civile) ha deciso di regolamentare i multicotteri in base alle caratteristiche e alle prestazioni globali: peso, raggio d’azione e quota di tangenza sono soltanto alcuni dei parametri presi in considerazione.

Privo di conducente, il drone è capace di compiere determinati tipi di missione tra due punti grazie ad un sistema di comando remoto ed è per questo motivo che viene denominato come APR (Aeromobile a Pilotaggio Remoto).

A seconda del raggio d’azione può essere dotato, per grandi distanze, di una videocamera con la

quale è possibile seguirne in tempo reale i movimenti in prima persona: questa classe viene chiamata FPV (First Person View).

Il drone è diventato così il velivolo di tutti i giorni utilizzato per scopi ludici come parimenti per scopi professionali, vedi riprese aeree e rilevamenti, o addirittura incursione aerea a corto raggio.

La sua varietà di scopi ne definisce da subito le caratteristiche e le soluzioni costruttive.

Varietà di scopi che si basano sui numerosi vantaggi di un velivolo ad ala rotante rispetto ad un modello ad ala fissa, a partire dal vantaggio in termine economico.

Appaiono poi fondamentali la notevole facilità di pilotaggio dovuti alle evoluzioni tecnologiche della microelettronica e dei sistemi di controllo oramai affidabili che rendono possibili varietà di missioni obbligatoriamente svolte in precedenza da elicotteri con pilota umano.

Non trascurabile la possibilità di volare in ambienti chiusi rendendolo decisamente più sicuro nel momento in cui ci si troverebbe invece in un posto angusto per una persona.

Per questa serie di motivi negli ultimi anni molta ricerca è stata effettuata in tutto il mondo riguardo simulazioni e controllo di piattaforme di questo tipo spesso adattate ad un compito in particolare da far assolvere al multirobot. Una di queste riguarda la capacità di un drone di essere vettore di trasporto di un carico non necessariamente ancorato ad un punto della scocca.

Realizzando sistemi di questa forma ed implementando sistemi di controllo si sarà in grado di svolgere missioni dove in precedenza venivano utilizzati elicotteri con equipaggio.

Esempi sono l'analisi del campo magnetico terrestre dove veniva utilizzato un elicottero con un magnetometro sospeso o lo studio di fondali marini sempre svolto con elicottero con payload immerso in acqua.

Struttura di un quadricottero:

Si può quindi definire il drone come un aerogiro radiocomandato a tutti gli effetti.

Come già introdotto esistono plurime soluzioni costruttive: tricottero, quadricottero, esacottero ed ottocottero votati semplicemente a diverse condizioni di carico necessario.

Per quanto diversi, il principio di funzionamento è simile per tutti i modelli, quindi, si presuppone che tutta la componentistica sia affine.

Partiamo dal telaio, lo scheletro del nostro mezzo, con il compito di sostenere il peso e resistere alle sollecitazioni nel corso di tutta la vita operativa.

Nello studio in dettaglio ci si è indirizzati sui quadricotteri, i più comuni sul mercato.

Di svariate forme, in un quadricottero esso ha una base centrale da cui si estenderanno tanti bracci quanti sono i motori adottati, perciò quattro bracci.

L'aspetto fondamentale su cui si insiste è lo stesso che in qualunque tipo di struttura: rigidità e peso. Se il telaio fosse troppo pesante il payload disponibile sarebbe notevolmente inferiore o richiederebbe una motorizzazione superiore per lo spostamento e questo influirebbe, in più, negativamente su costi e consumi.

Se il materiale fosse invece eccessivamente elastico le forze agenti alle estremità dei bracci, dovute al tiro delle eliche, deformerebbero troppo la struttura causando uno spostamento delle forze dalla perpendicolare al piano appartenente al quadricottero.

Ne consegue una maggiore instabilità nelle fasi più cruciali del volo o deviazioni dalla traiettoria prevista.

Un telaio leggero e rigido in alluminio o materiale composito permette di risparmiare peso, così da incrementare il carico pagante, o di necessitare di meno energia per la stessa manovra aumentando l'autonomia in volo, uno dei punti focali per il futuro sviluppo dei droni.

L'autonomia è espressa dalla capacità delle batterie installate che sono sempre posizionate al centro della struttura: coprendo una grossa percentuale del peso complessivo posizionarle nei pressi del baricentro aumenta molto la stabilità e ne diminuisce l'inerzia delle manovre.

Attualmente vengono utilizzate quelle a Litio-Polimero (LiPo) per una serie di vantaggi su molte delle formule più comuni in commercio come Litio-Ione (LiIon) o Nichel-Cadmio (NiCd). La mancanza di un solvente organico all'interno le rende innanzitutto più sicure se danneggiate, essendo costituite di un materiale solido: appunto il polimero. Il materiale costituente inoltre non richiede un involucro metallico, per cui sono più leggere, e ha una densità di energia maggiore avendo un impacchettamento più denso. La capacità di essere sagomata a piacimento e un maggiore

ciclo di vita infine ne hanno fatto la scelta più vantaggiosa. Possono essere disposte in serie o parallelo, atte a tenere in vita i sistemi di controllo del drone e di raccolta dati. Il primo utilizzatore è sicuramente l'apparato propulsivo. Quest'ultimo è formato da motori, ingranaggi, eliche.

I motori sono elettrici, maggiormente economici e gestibili rispetto a quelli a pistoni.

Tuttavia scartiamo i motori trifase per le dimensioni e il peso, e quelli a spazzole per i problemi di scintillio dovuti agli alti regimi di rotazione. Restano quindi i motori elettrici brushless (senza spazzole) a cassa rotante più adatti all'utilizzo in questione.

L'energia è fornita dalle batterie ma il giusto voltaggio di cui i propulsori necessitano è garantito dalla presenza di regolatori (ESC). Questi componenti, uno per ciascun rotore, agiscono tramite un input del controllore che da loro l'informazione sul voltaggio corretto.

La rotazione impressa dai motori viene trasmessa tramite ingranaggi con un determinato rapporto di marcia alle eliche.

La scelta dell'elica deve essere fatta in modo tale da trovare il corretto punto di funzionamento con il motore scelto, quindi nei regimi dove viene espressa la potenza massima.

Se così non fosse, l'elica, per quanto prestazionale, non lavorerebbe con la giusta efficienza: in termini pratici un'elica troppo leggera porterebbe il motore fuori giri, in caso contrario, una troppo pesante non riuscirebbe ad essere sufficientemente trascinata. Spinta e spostamento in ogni direzione sono perciò ottenuti da eliche collegate ai propulsori che generano variazioni di portanza.

Eliche a passo fisso sono più comuni per il costo e la semplicità costruttiva ma sono disponibili anche soluzioni più complesse con l'utilizzo di eliche a passo variabile: nelle prime le variazioni di portanza si hanno variando i giri del motore mentre nelle seconde modificando l'incidenza delle pale rispetto al flusso. Soluzioni economiche sono costruite in polimero ma si possono avere anche eliche in fibra di carbonio.

Il funzionamento corretto di tutti questi organi è coadiuvato dal controllore del nostro drone.

Anch'esso posizionato al centro per le dimensioni che questo elemento occupa, il controllore non è altro che una scheda programmabile, molto spesso di tipo Arduino.

Su di essa è montato un processore per l'analisi degli dati dai vari input provenienti da piattaforme inerziali (IMU), GPS, altimetro e magnetometro.

La IMU (Inertial Measurement Unit) misura accelerazioni e velocità angolari tramite accelerometri e giroscopi che registrano posizione e velocità. Il GPS permette la localizzazione del nostro velivolo mentre l'altimetro registra valori sull'altitudine.

Infine il magnetometro è una bussola che analizza la posizione del Nord e permette di conoscere l'orientamento del quadricottero rispetto ad esso. Note queste informazioni il processore invia ai regolatori il segnale per il corretto voltaggio da fornire alla motorizzazione utilizzata.

L'interfacciamento con l'utente avviene mediante il radiocomando, o meglio la trasmittente, e il ricevitore. Quest'ultimo riceve i segnali del pilota come un segnale radio e successivamente li converte in impulsi elettrici per il controllore.

Cap.1: Fisica di un Quadricottero Multirotore

1.1 Assetto e Meccanica del Volo:

Per quanto concerne la meccanica del volo di un quadricottero, si è posta l'attenzione su droni ad ala rotante con passo fisso. E' comprensibile che la scelta del quadricottero in senso stretto appaia come una soluzione efficace in chiave di controllabilità e stabilità dovuta al simmetrico posizionamento dei motori a croce, in genere con assi quasi perpendicolari.

Malgrado ciò, il sistema nel complesso risulta sotto-attuato: il drone infatti possiede sei gradi di libertà (6 DoF), dei quali tre sono dovuti alla rotazione intorno a tre assi mentre i restanti tre al movimento in tre direzioni, ma può controllare il suo movimento soltanto con i quattro attuatori.

Il rollio indica la rotazione intorno ad un asse longitudinale del corpo (roll), il beccheggio la rotazione intorno ad un asse trasversale del corpo (pitch), l'imbardata denota infine la rotazione intorno ad un asse verticale passante per il baricentro del corpo (yaw).

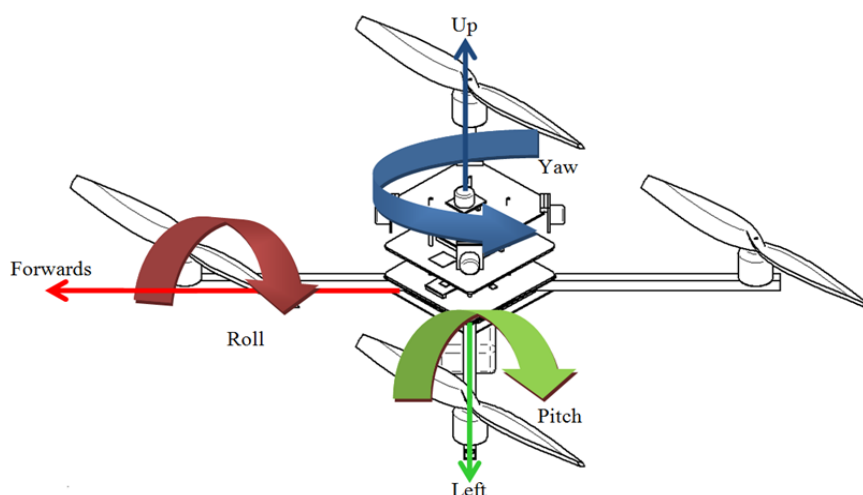


Fig.2: Rappresentazione dei movimenti intorno agli assi.

Lo spostamento nelle tre direzioni si suddivide in spostamento lungo la direzione frontale del mezzo (x), lungo la direzione laterale del veicolo (y) e spostamento in altezza (z).

Dei quattro propulsori due ruotano in verso orario due in verso antiorario: in particolare su ciascun braccio della croce i due motori opposti ruotano equiversi (1-3 e 2-4).

Su ciascuna diagonale quindi si viene ad avere una coppia agente sull'asse su cui sono posizionati i motori.

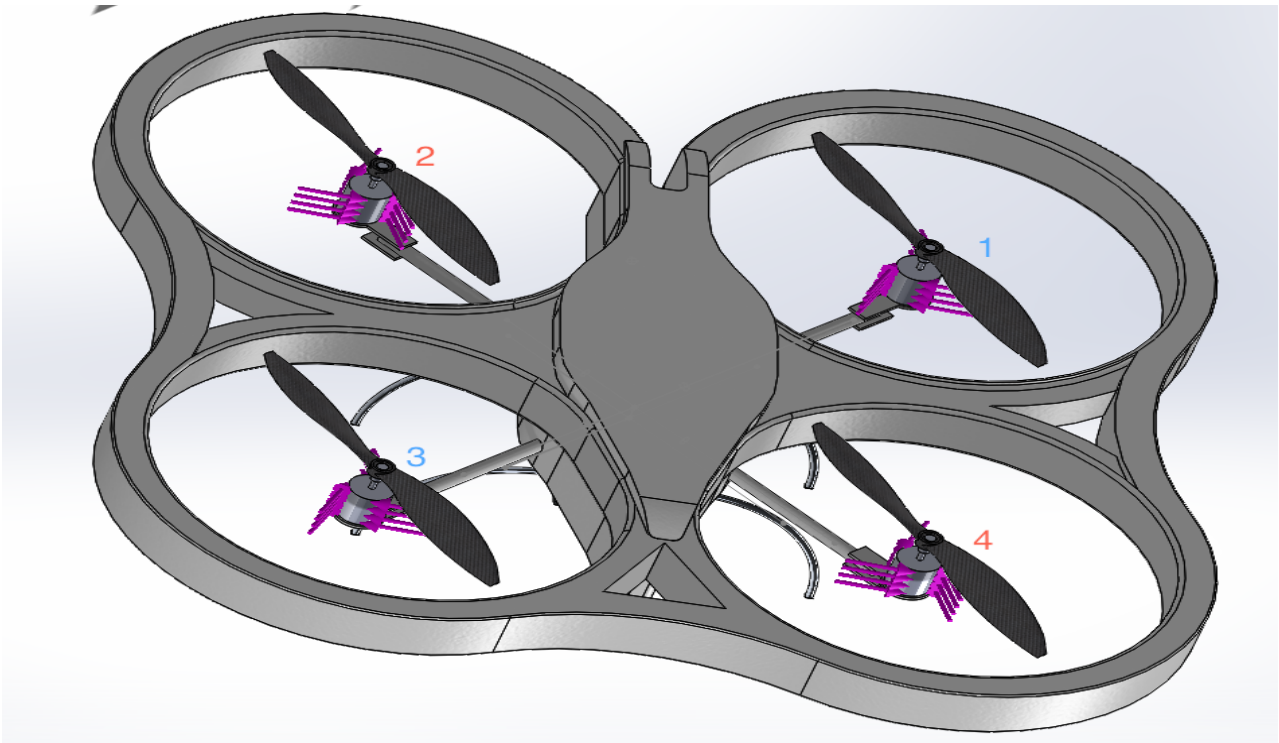


Fig.3: Modello delle coppie agenti su un quadrotore.

Se tutti i motori girano ad un valore di giri identico, non c'è necessità di rotori aggiuntivi per equilibrare le coppie generate, in quanto uguali ed opposte, ottenendo accelerazione angolare nulla intorno all'assi di rotazione.

Ne consegue che modificando opportunamente i giri, quindi le forze, si ottiene uno scompenso di forze e/o momenti in grado di far effettuare al veicolo le manovre desiderate.

Nel dettaglio, la manovra di salita, la più semplice viene eseguita aumentando la spinta di tutti i rotori allo stesso modo.

La condizione di volo a punto fisso, detta anche hovering, è possibile considerarla un caso particolare della manovra di salita in cui ciascun motore ha tiro pari ad un quarto del peso del drone raggiungendo il bilancio delle forze tra spinta e peso.

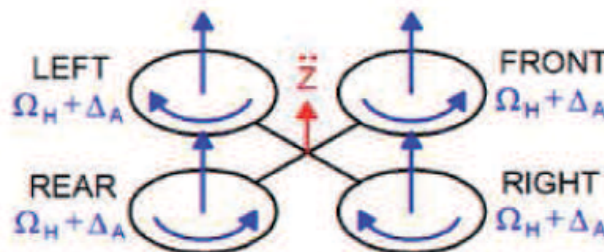


Fig.4: Manovra di salita.

Movimenti attorno agli assi di rollio o beccheggio possono essere effettuati senza influenzare l'asse di imbardata. Più precisamente la tecnica adottata per il rollio e il beccheggio è la stessa ma si agisce su coppie di motori diversi: quelli sull'asse longitudinale (x) per il beccheggio e quelli sull'asse laterali per il rollio (y).

Incrementando la potenza di un rotore e decrementando quella dell'opposto si mantengono le coppie complessive globalmente nulle ma si provoca una rotazione attorno dell'asse su cui i motori non cambiano regime: il laterale per il beccheggio (y) e il longitudinale per il rollio (x), ovvero le manovre desiderate.

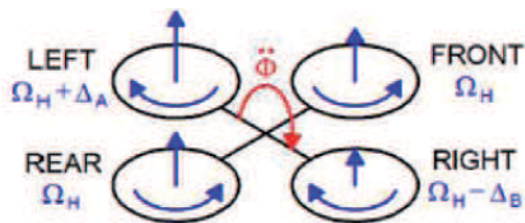


Fig.5: Manovra di rollio (roll).

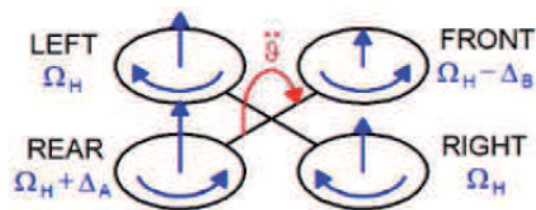


Fig.6: Manovra di beccheggio (pitch).

Riguardo al movimento di imbardata, esso è ricavato fornendo potenza ugualmente su una coppia di motori e togliendone allo stesso modo sull'altra.

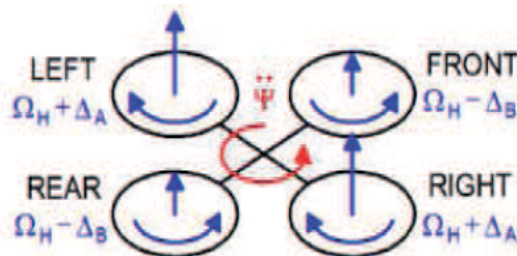


Fig.7: Manovra di imbardata (yaw).

La portanza complessiva è così mantenuta da tiro totale dei rotori ma si provoca uno squilibrio dei momenti che genera una rotazione intorno all'asse verticale.

La trattazione è valida con dovuti accorgimenti anche per eliche a passo variabile: cambia, come già spiegato, il modo in cui viene ottenuta una differenza di forza e di momento.

1.2 Angoli di Eulero:

Gli angoli di Eulero sono uno strumento per avere una descrizione di un corpo rigido nello spazio.

In particolare descrivono gli sfasamenti angolari di una terna di riferimento solidale attraverso una serie di rotazioni elementari a partire da un sistema di riferimento fisso.

Nella rappresentazione sono assunte due terne, una fissa xyz ed una ad essa solidale XYZ .

Vengono così definiti:

ϕ : chiamato angolo di rotazione propria, generato facendo ruotare la terna XYZ intorno all'asse X ;

θ : chiamato angolo di nutazione, ottenuto con la rotazione attorno all'asse Y ;

ψ : chiamato angolo di precessione, realizzato con la rotazione questa volta intorno all'asse Z .

Il senso della rotazione viene considerato positivo in senso antiorario.

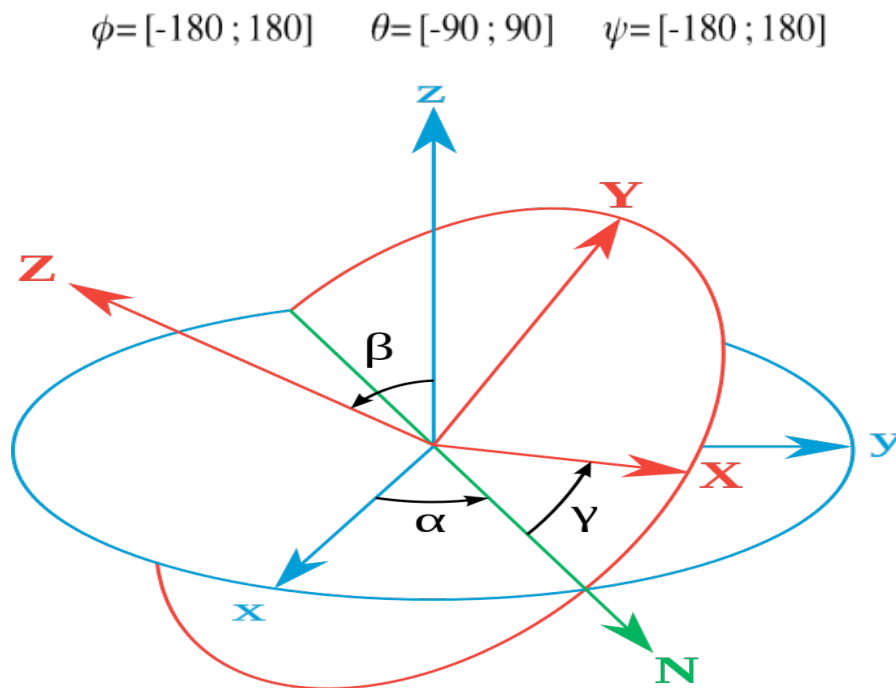


Fig.8: Rappresentazione classica degli angoli di Eulero

A queste rotazioni sono paragonabili le singole manovre che il mezzo può effettuare, in particolare ad'un inclinazione attorno ad un asse si può associare uno dei angoli di Eulero prima definiti:

- alla rotazione attorno all'asse longitudinale del corpo daremo il valore ϕ (roll);
- alla rotazione attorno all'asse trasversale daremo il valore θ (pitch);
- alla rotazione attorno all'asse verticale passante per il baricentro il valore ψ (yaw).

1.3 Modello Matematico:

Prima di cominciare con l'implementazione vera e propria, fondamentale è ottenere un modello matematico in grado di interpretare e modellare in maniera corretta gli input forniti.

Vengono ivi illustrate le tecniche e le equazioni utilizzate, con opportune semplificazioni, per descrivere il comportamento complessivo del quadricottero.

1.3.1 Ambiente Matematico:

Conoscendo il comportamento in volo e definite le rotazioni elementari, il primo step consiste quindi nel posizionare correttamente la terna di riferimento fissa e quella solidale in modo tale da ricavarne posizione e orientamento del velivolo in maniera più semplice possibile.

Il sistema di riferimento fisso (SdR), inerziale, è scelto in riferimento al piano tangente passante per la superficie terrestre in ogni punto: è stata chiamata O_E .

La appendice "E" sta per Earth, da cui i tre assi riprendono le direzioni, con gli assi x ed y appartenenti al piano tangente che puntano rispettivamente verso il Nord e l'Est, l'asse z invece che ha direzione perpendicolare al piano e diretto verso l'alto.

La terna mobile viene invece presa solidale con il baricentro del quadricottero ed è presa dalla letteratura, nominata O_{ABC} (Aircraft Body Center). Entrambe i SdR sono sinistrorsi ed è necessario adottarle entrambe per risolvere efficacemente le equazioni del modello.

Per eseguire il passaggio dal SdR mobile O_{ABC} a quello fisso O_E intervengono tre matrici di rotazione, ciascuna corrispondente ad un angolo, e quindi ad una manovra di roll, pitch o yaw.

Le matrici sono così definite:

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$R_z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Una matrice totale di rotazione in R^3 , nota anche come DCM (Direction Cosine Matrix), si ottiene moltiplicando le matrici prima illustrate.

$$DCM(\phi, \theta, \psi) = \begin{bmatrix} \cos\theta \cos\psi & \cos\phi \sin\psi + \sin\phi \sin\theta \cos\psi & \sin\phi \sin\psi - \cos\phi \sin\theta \cos\psi \\ -\cos\theta \sin\psi & \cos\phi \cos\psi - \sin\phi \sin\theta \sin\psi & \sin\phi \cos\psi + \cos\phi \sin\theta \sin\psi \\ \sin\theta & -\sin\phi \cos\theta & \cos\phi \cos\theta \end{bmatrix}$$

1.3.2 Equazioni di Eulero-Newton:

In questo capitolo ci si propone di ottenere le espressioni matriciali ottenendo la modellizzazione di un caso quasi realistico. L'assetto e le posizioni di un sistema di movimento a sei gradi di libertà viene descritta attraverso dodici equazioni. Le variabili interessate appartengono quindi sei al sistema di riferimento O_E , sei al sistema O_{ABC} .

Si introducono, quindi, i vettori che richiamano le velocità di traslazione (V_b) e di rotazione (ω_b) espressi nella terna di riferimento del sistema corpo del drone.

$$V_b = [u \quad v \quad w]^T$$

$$\omega_b = [p \quad q \quad r]^T$$

Nel sistema di riferimento degli assi terrestri invece introduciamo il vettore lineare posizione (P_e) e il vettore che rappresenta l'orientamento nello spazio del velivolo tramite gli angoli di Eulero (A_e).

$$P_e = [x \quad y \quad z]^T$$

$$A_e = [\phi \quad \theta \quad \psi]^T$$

Di conseguenza è possibile definire:

$$\dot{P}_e = DCM \cdot V_b$$

$$\dot{A}_e = ROT \cdot \omega_b$$

Dove ROT è una matrice in R^3 atta alle trasformazioni angolari parimenti alla DCM precedente.

$$ROT = \begin{bmatrix} 1 & \sin\phi \, tg\theta & \cos\phi \, tg\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{ctg\theta} & \frac{\cos\phi}{ctg\theta} \end{bmatrix}$$

Esplicitando i prodotti matrciali si ha:

$$\begin{cases} \dot{x} = u \cdot \cos\theta \cos\psi + v (\sin\theta \sin\phi \cos\psi - \cos\phi \sin\psi) + w (\sin\theta \cos\phi \cos\psi + \sin\phi \sin\psi) \\ \dot{y} = u \cdot \cos\theta \sin\psi + v (\sin\theta \sin\phi \sin\psi + \cos\phi \cos\psi) + w (\sin\theta \cos\phi \sin\psi - \sin\phi \cos\psi) \\ \dot{z} = -u \sin\theta + v \cos\theta \sin\phi + w \cos\theta \cos\phi \end{cases}$$

$$\begin{cases} \dot{\phi} = p + q \cdot tg\theta \cdot \sin\phi + r \cdot tg\theta \cdot \cos\phi \\ \dot{\theta} = q \cdot \cos\theta - r \cdot \sin\phi \\ \dot{\psi} = \frac{\sin\psi}{\cos\theta} \cdot q + \frac{\cos\psi}{\cos\theta} \cdot r \end{cases}$$

Rilevante è che nel set di equazioni rappresentanti gli angoli di Eulero il calcolo dell'angolo ψ è inscindibile dagli altri due. Nella pratica questo significa che manovre in rollio (ϕ) e/o beccheggio (θ) possono automaticamente causare variazioni angolari di ψ ottenendo anche una manovra di imbardata. Queste rappresentano le nostre prime sei espressioni espresse nella terna fissa terrestre. Una volta introdotte queste trasformazioni, sotto specifiche ipotesi, è applicabile la seconda legge di Newton.

E' necessario innanzitutto che l'origine di O_{ABC} coincida con il baricentro del quadricottero; in più

gli stessi assi di O_{ABC} devono coincidere con gli assi principali di inerzia.

Con quest'ultima assunzione si ottiene un tensore di inerzia diagonale senza la presenza di termini dovuti alle accelerazioni centrifughe. Esplicitando le leggi cardinali della dinamica:

$$m \cdot \ddot{P}_e = \sum_1^n F_i$$

$$I \cdot \ddot{A}_e = \sum_1^n \tau_i$$

Dove I indica il tensore d'inerzia diagonale.

Adesso, riutilizziamo le derivate prime della traslazione all'interno della prima espressione:

$$m \cdot \frac{\partial}{\partial t}(DCM \cdot V_b) = DCM \cdot F_b$$

Sviluppando si ha:

$$m \cdot (DCM \cdot \dot{V}_b + DCM \cdot \dot{V}_b) = DCM \cdot F_b$$

Ricordando che la velocità di un SdR è pari alla velocità lineare della terna più la sua stessa rotazione:

$$m \cdot DCM \cdot (\dot{V}_b + \omega_b \times V_b) = DCM \cdot F_b$$

Con una facile semplificazione:

$$m \cdot (\dot{V}_b + \omega_b \times V_b) = F_b$$

Questa è la forma generica della equazione di Newton.

Si ripercorrono ora gli stessi passaggi utilizzando le derivate prime questa volta delle rotazioni all'interno della seconda legge della dinamica:

$$I \cdot \frac{\partial}{\partial t} (ROT \cdot \omega_b) = ROT \cdot \tau_b$$

Si trova così

$$I \cdot \frac{\partial}{\partial t} (\dot{ROT} \cdot \omega_b + ROT \cdot \dot{\omega}_b) = ROT \cdot \tau_b$$

$$I \cdot ROT \cdot \dot{\omega}_b + ROT \cdot \omega_B \times I \cdot \omega_b = ROT \cdot \tau_b$$

Semplificando infine:

$$I \cdot (\dot{\omega}_b + \omega_b \times I \cdot \omega_b) = \tau_b$$

Questa, invece, rappresenta l'equazione di Eulero. F_b e τ_b rappresenta nelle nostre equazioni finali le forze agenti sulla scocca del nostro corpo e il risultante delle coppie espressi nel riferimento della terna O_{ABC} . Le due equazioni vettoriali sono traducibili quindi in sei equazioni scalari.

Vengono espresse sotto forma di prodotti tra matrici:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = m \begin{bmatrix} \dot{u} + qw - rv \\ \dot{v} + ru - pw \\ \dot{w} + pv - qu \end{bmatrix} = F_{body}$$

$$\begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = m \begin{bmatrix} \dot{p} I_x + qr (I_z - I_y) \\ \dot{q} I_y + pr (I_x - I_z) \\ \dot{r} I_z + pq (I_y - I_x) \end{bmatrix} = \tau_{body}$$

In questo modo si possiedono gli strumenti per il calcolo di tutte le informazioni sul nostro sistema in movimento.

1.4 Momento di Inerzia:

Per poter applicare gli agenti esterni, si necessita dapprima del calcolo del baricentro e dei momenti di inerzia per poter definire le proprietà di massa del drone.

Logicamente, il calcolo di queste grandezze è imprescindibile dal modello considerato e quindi dalle sue caratteristiche costruttive. Il veicolo scelto, è un quadricottero realizzato in ambiente CAD Solidworks.

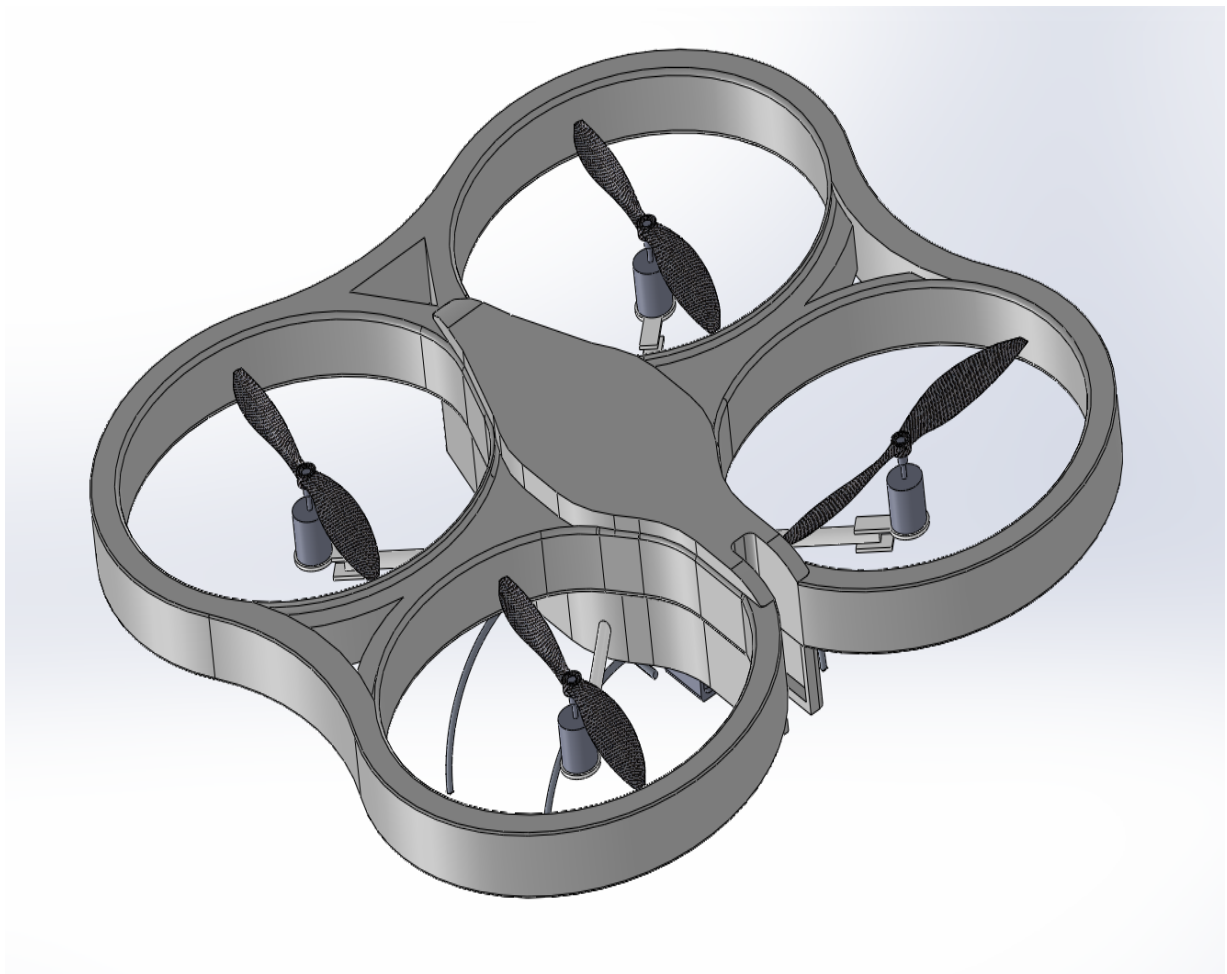


Fig.9: Visione completa dell'assieme.

Componente	Materiale (Ove Presente)	Massa/Densità Inserita	Peso
Guscio Sup. Est.	Polistirolo/Carbonio	Densità	1635.0 g
Guscio Inf. Est.	Polistirolo/Carbonio	Densità	124.6 g
Vano Centrale	HDPE	-	1740.9 g
4x Motore Elettrico	-	Massa	2380.0 g
4x Elica	Carboresina	Massa	650.0 g
Gambe Appoggio	Alluminio DIN 3.0205	-	475.5 g
Telemetro Laser	-	Massa	220.0 g
Variatore (con cavi)	-	Massa	155.0 g
4x Batteria	-	Massa	2640.0 g
PESO TOTALE			10023.0 g

Fig.10: Componenti del quadricottero.

Benché allo stato di progettazione preliminare, sono presenti tutti gli elementi fondamentali per il suo funzionamento. La possibilità inoltre di poter assegnare dei materiali, o comunque una densità, ai vari componenti ne permettono una stima più precisa di un calcolo semplificato effettuato in maniera manuale.

Inoltre tramite lo strumento CAD bisogna solamente leggere in output il valore da lui restituito in maniera automatica una volta creato il sistema di riferimento tramite “Geometria di riferimento”.

Non essendo perfettamente simmetrico (i bracci anteriori sono leggermente più lunghi di quelli posteriori), la matrice d’inerzia non risulta perfettamente diagonale:

Proprietà di massa di QUAD SIMULINK		
Configurazione: Default		
Sistema di coordinate: Sistema di coordinate 1		
Massa = 10.0237 chilogrammi		
Volume = 0.0551 metri cubici		
Area superficie = 3.8516 metri quadrati		
Centro di massa: (metri)		
X = -0.0351		
Y = 0.0875		
Z = 1.0757		
Asse principale di inerzia e momenti principali di inerzia: (chilogrammi * metri quadrati)		
Nel centro della massa.		
Ix = (0.1365, 0.9906, 0.0049)		Px = 0.3852
Iy = (-0.9906, 0.1365, -0.0046)		Py = 0.3988
Iz = (-0.0052, -0.0042, 1.0000)		Pz = 0.7335
Momenti di inerzia: (chilogrammi * metri quadrati)		
Presi nel centro di massa e allineati con il sistema di coordinate risultato.		
Lxx = 0.3986	Lxy = 0.0018	Lxz = 0.0018
Lyx = 0.0018	Lyy = 0.3854	Lyz = 0.0015
Lzx = 0.0018	Lzy = 0.0015	Lzz = 0.7335

Fig.11: I momenti di inerzia ottenuti con il nuovo sistema di coordinate.

Tuttavia con un approssimazione i termini che rappresentano i momenti deviatori sono due ordini di grandezza inferiori e possono essere considerati trascurabili.

In questo modo si ha un tensore di inerzia ideale, ovvero applicato sugli assi principali di inerzia del corpo in questione e in particolare solidali con gli assi corpo.

1.5 Forze e Momenti Agenti:

Alla formulazione precedente devono essere applicate le forze e i momenti esterni agenti sul velivolo. Ciò che deve essere analizzato nel dettaglio è il modo di cui questi elementi si compongono. Riguardo alle forze, possiamo scomporle in forza peso, dovuta all'attrazione gravitazionale della Terra, e in forza dei motori, che in gergo possiamo definire "spinta".

Per il peso si utilizza la formulazione classica: il prodotto massa moltiplicata per la gravità.

La direzione di questo vettore, ovviamente, sarà sempre verticale e diretta verso il basso indipendentemente dall'orientamento del corpo.

Per la spinta dei motori si considera una modellazione semplificata dipendente dal quadrato delle velocità di rotazione di ciascun rotore:

$$T = b \left(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2 \right)$$

La costante b rappresenta il coefficiente di spinta dei motori ed è un dato del nostro problema.

Essendo il tutto calcolato in assi corpo, la direzione della spinta risulterà sempre perpendicolare al piano contenente il drone. Per il peso, il vettore avrà componenti diverse variabili invece con le possibili rotazioni del quadricotore, in particolare con la matrice DCM.

Le forze esterne quindi sono sintetizzabili in:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = F_{ext} = DCM \cdot \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix}$$

L'analisi delle coppie generate è stata effettuata guardando l'influenza di ciascun tipo di manovra intorno ad un asse a partire dalla forza generata come per la spinta verticale.

Moltiplicando le forze per il braccio agente, ovvero la distanza tra punto di applicazione della forza e baricentro corrispondente per approssimazione al semiasse della scocca, si ottengono momenti agenti su diversi assi.

Anche in questo caso è evidente la dipendenza quadratica del valore delle coppie al variare dei giri.

$$u_2 = b \left(\omega_4^2 - \omega_2^2 \right)$$

$$u_3 = b \left(\omega_3^2 - \omega_1^2 \right)$$

$$u_4 = d \left(\omega_2^2 + \omega_4^2 - \omega_1^2 - \omega_3^2 \right)$$

La costante presa in considerazione questa volta è anche b, costante di coppia della motorizzazione scelta, ottenibile allo stesso modo di d.

In quanto ogni manovra agisce con una rotazione attorno ad un asse, parimenti ad un momento, il

vettore che rappresenta i momenti esterni può essere scritto in funzione delle manovre stesse.

$$M_x = l \cdot u_2 = l \cdot b \left(\omega_4^2 - \omega_2^2 \right)$$

$$M_y = l \cdot u_3 = l \cdot b \left(\omega_3^2 - \omega_1^2 \right)$$

$$M_z = l \cdot u_4 = l \cdot d \left(\omega_2^2 + \omega_4^2 - \omega_1^2 - \omega_3^2 \right)$$

1.6 Dati e Input del Sistema:

Conosciuto l'ambiente e la sua impostazione matematica, il quadricottero e le equazioni risolutive del moto ad esso relativo si devono definire quelli che rappresentano gli input del nostro problema:

- dati relativi al mezzo: peso, semi-diagonale del telaio, coefficienti delle motorizzazioni, matrice d'inerzia prima ottenuta;
- dati relativi alle condizioni iniziali del mio problema: i vettori posizione (P_e) e gli angoli di Eulero (A_e) nella terna fissa terrestre (O_E) e vettori velocità di traslazione (V_b) e velocità angolare (W_b) nel sistema solidale al baricentro (O_{ABC});
- le forze e i momenti esterni definiti all'istante iniziale.

1.7 Modello Utilizzato:

Scopo ultimo dell'ambiente matematico costituito è quello di ottenere una simulazione temporale in un intervallo di tempo prefissato dove vengono analizzate tutte le variabili che rappresentano il moto del quadricottero: in sintesi, tutti i vettori prima introdotti, A_e , P_e , V_b , W_b .

Non implementando ora alcun sistema di controllo, ma solo una descrizione semplificata del moto, è corretto pensare di introdurre come dato noto fattori esterni agenti sul velivolo anziché averli come incognite, così da ricavarne dalle equazioni di Newton ed Eulero le grandezze relative agli assi corpo (V_b e W_b).

Tuttavia è da sottolineare che ogni input da noi introdotto non possiede alcun transitorio e rimane costante per tutto l'intervallo della simulazione.

In aggiunta non è stata inserita alcuna forza di resistenza quindi in presenza di una forza si assisterà ad una accelerazione continua che seguirà l'attitudine del velivolo con le proprie condizioni iniziali. Ciò che ha davvero rilevanza sono i dati nei primi momenti, ovvero lo studio di fattibilità della manovra e non la sua dinamica.

Per effettuare questa operazione è necessario invertire le equazioni di Newton-Eulero, avendo quindi come incognite le derivate prime temporali di velocità di traslazione e velocità angolare.

Di seguito:

$$\begin{cases} \dot{p} = \frac{1}{I_x} \cdot \{ M_x - [qr (I_z - I_y)] \} \\ \dot{q} = \frac{1}{I_y} \cdot \{ M_y - [pr (I_x - I_z)] \} \\ \dot{r} = \frac{1}{I_z} \cdot \{ M_z - [pq (I_y - I_x)] \} \end{cases}$$

$$\begin{cases} \dot{u} = F_x + m (qw - vr) \\ \dot{v} = F_y + m (ru - pw) \\ \dot{z} = F_z + m (pv - qu) \end{cases}$$

Successivamente si può ottenere una soluzione integrata in un intervallo a nostra scelta per seguirne l'andamento. La soluzione integrata non è una funzione continua ma un insieme di punti presi in vari istanti di tempo equispaziati.

Il risultato di questa prima parte è espresso quindi dai due vettori V_b e W_b dipendenti dal tempo sotto forma di un insieme discreto di valori. Dopo le equazioni della cinematica si sfruttano parimenti quelle della dinamica. Dapprima vengono presi in successione due valori di V_b e W_b rispettivamente per ogni istante di tempo e si calcolano progressivamente tramite le matrici di rotazione i corrispettivi valori derivati di A_e e P_e , orientamento e posizione nello spazio.

Si otterranno tanti valori di A_e e P_e derivati tanti quanti saranno gli intervalli di tempo stabiliti.

Il passaggio seguente consiste nell'integrare alla stessa maniera questi numeri in istanti di tempo minori così da ottenere una sorta di integrazione dell'integrazione.

Riutilizzando le equazioni:

$$\begin{cases} \dot{x} = u \cdot \cos\theta \cos\psi + v (\sin\theta \sin\phi \cos\psi - \cos\phi \sin\psi) + w (\sin\theta \cos\phi \cos\psi + \sin\phi \sin\psi) \\ \dot{y} = u \cdot \cos\theta \sin\psi + v (\sin\theta \sin\phi \sin\psi + \cos\phi \cos\psi) + w (\sin\theta \cos\phi \sin\psi - \sin\phi \cos\psi) \\ \dot{z} = -u \sin\theta + v \cos\theta \sin\phi + w \cos\theta \cos\phi \end{cases}$$

$$\begin{cases} \dot{\phi} = p + q \cdot \tan\theta \cdot \sin\phi + r \cdot \tan\theta \cdot \cos\phi \\ \dot{\theta} = q \cdot \cos\theta - r \cdot \sin\theta \\ \dot{\psi} = \frac{\sin\psi}{\cos\theta} \cdot q + \frac{\cos\psi}{\cos\theta} \cdot r \end{cases}$$

Avremo, di conseguenza, una successione discreta di valori molto più estesa e dettagliata dei dati rispetto agli assi fissi che a quelli solidali al baricentro del quadrirotore.

Tappa conclusiva di questa procedura è stata ottenere tramite valori finiti degli andamenti reali, seppur approssimati, di possibili configurazioni di volo.

Cap.2: Implementazione Modello e Simulazioni

Matlab e Simulink:

L'ambiente Matlab permette di lavorare in ambienti matematici complessi gestendo i dati come se questi fossero matrici. La presenza di strumenti specializzati, tramite librerie, per ogni tipo di approccio ingegneristico ne rende lo strumento ideale per la risoluzione di calcoli numerici.

Simulink è un'estensione di Matlab che tramite l'adozione di interfaccia per l'utente tramite schema a blocchi modella e simula come Matlab stesso sistemi di ogni tipo.

Si è scelto perciò di adottarli per la risoluzione delle equazioni del moto da noi utilizzate e per simulare manovre di base effettuando un confronto tra i due strumenti messi a disposizione.

Si tratta di un sistema che presenta forte dipendenza dalle funzioni trigonometriche quindi è sicuramente di tipo non lineare.

2.1 Implementazione Codice Matlab:

L'implementazione in Matlab si svolge in diversi step seguendo circa la linea del modello matematico non lineare.

Il primo passo consiste con l'introduzione di tutti i dati e i valori di ingresso necessari per cominciare la risoluzione.

Questi, illustrati già in precedenza, sono introdotti direttamente all'interno dello script o facilmente calcolati in fase iniziale: forze e momenti esterni sono trovati in questa fase.

Dipenderanno ovviamente dal tipo di manovra che si vorrà simulare.

Benché ci si occupi di manovre semplici, la fattibilità di manovre complesse può essere comunque visualizzata nella loro evoluzione fino al tempo di considerazione voluto allo stesso modo.

Seguono, le due matrici di rotazione, DCM e ROT, prelievate condizioni iniziali di assetto, dalle quali dipendono matematicamente. Nello sviluppo seguente le equazioni descriventi velocità di traslazione e velocità angolare in assi corpo, rispettivamente V_b e W_b , sono trascritte scalarmente tre per tre ed espresse in sei corrispondenti funzioni.

Le variabili delle function sono relative ai dati introdotti a monte e sono ivi utilizzati.

```
function DU = risoluzione_U (t,U,Fx,m,q,r,v,w)
function DV = risoluzione_V (t,V,Fy,m,u,w,p,r)
function DW = risoluzione_W (t,W,m,p,q,u,v,Fz)
```

Queste risolvono le velocità di traslazione calcolate in O_{ABC} .

```
function DP = risoluzione_P (t,P,l,r,q,Ix,Iy,Iz,om2,om4,b)
function DQ = risoluzione_Q (t,Q,Ix,Iy,Iz,p,l,r,om3,om1,b)
function DR = risoluzione_R (t,R,Ix,Iy,Iz,p,q,b,l,d,om1,om2,om3,om4)
```

Queste risolvono invece le velocità di rotazione calcolate sempre in O_{ABC} .

Insieme alle function si adopera un risolutore di funzioni differenziali ordinarie, dette ODE, tra quelle di default nella libreria di Matlab chiamata “ode45”.

Questo risolutore utilizza in particolare il metodo di Runge-Kutta, spesso abbreviato RK, formato da una famiglia di iterazioni discrete adatte alle equazioni ODE di medio grado specificatamente per problemi ai valori iniziali.

La sintassi necessaria per l'utilizzo di “ode45” è la seguente:

```
options = odeset ('name1', value1, ..., 'nameN', valueN)
[T,X] = ode45 (odefun, tspan, x0, options, var1, ..., varN)
```

Viene utilizzato un risolutore per ogni function richiamata tramite l'operatore “@” .

Di contorno vengono introdotte specifiche come l'intervallo di integrazione diviso equamente in un numero sufficiente di valori per ottenere una buona precisione nel risultato che sarà comune a tutte le funzioni. Le simulazioni saranno studiate da zero a dieci secondi divisi per intervalli in 0.01 s, per ottenere la buona precisione assoluta richiesta.

Le sei condizioni iniziali necessarie sono espresse nei vettori V_b e W_b dati inizialmente e sono espresse con le lettere in minuscolo (p,q,r per W_b e u,v,w per V_b). La presenza delle parentesi quadre vuote implica soltanto che l'implementazione è stata fatta non usando l'argomento “option”.

Le variabili che si susseguono infine sono le stesse che vengono chiamate all'interno di ciascuna function. I risultati di questa integrazione discretizzata possono essere salvati in formati diversi a seconda del trattamento che ne si vuole fare. Introducendo una variabile ad esempio "a" i dati vengono salvati in variabili di tipo struttura (struct).

Invece si è preferito introdurre un vettore tempo variante dove ad ogni istante di tempo considerato si ha un valore della funzione integrata. La forma utilizzata è un vettore a due righe espressa nella sintassi di "ode45".

Per allocare i risultati in quest'ultima forma, da notare, si necessita l'introduzione come incognite all'interno delle funzioni anche la variabile "t".

Essendo presenti sei risolutori "ode45" come output si avranno sei vettori tempo varianti che rappresentano l'andamento e quindi l'evoluzione dei valori delle variabili relativi alle terne in assi corpo. Rispettivamente i valori sono espressi in U,V,W per l'integrazione delle equazioni di V_b e in P,Q,R per quelle di W_b .

Si sono poi raccolti i valori U,V e W nella variabile V_{b-eff} e P,Q ed R in W_{b-eff} .

In questo momento quindi conosciamo tutti i dati relativi agli assi O_{ABC} relativi al corpo durante il tempo della simulazione.

Nella seconda parte si troveranno le grandezze relative al sistema di riferimento terrestre, O_E .

Il frutto dei passaggi precedenti è quindi racchiuso in V_{b-eff} e W_{b-eff} .

Avendo integrato funzioni costanti quelle che si sono ottenute sono funzioni di primo grado.

Per l'elaborazione di ciascuna si sono implementati due script aventi lo stesso funzionamento.

Avendo una moltitudine di valori per ogni grandezza rappresentati in vari istanti di tempo, quello che si è fatto è stato integrare all'interno di questi intervalli con intervalli ancora più piccoli.

Per l'integrazione il risolutore adottato è sempre "ode45" con tutte le sue specifiche di funzionamento.

Le funzioni chiamate questa volta sono le equazioni scalari che invertono i dati da O_{ABC} a O_E tramite le matrici di rotazione DCM per le velocità di rotazione e ROT per le velocità di traslazione.

I risultati saranno poi integrati come precedentemente.

```
function DX=risoluzione_X (t,X,DCM,Vb_eff)
function DY=risoluzione_Y (t,Y,DCM,Vb_eff)
function DZ=risoluzione_Z (t,Z,DCM,Vb_eff)
```

Danno come output la posizione del velivolo espressa in O_E .

```
function DPHI=risoluzione_PHI (t,PHI,ROT,Wb_eff)
function DPSI=risoluzione_PSI (t,PSI,ROT,Wb_eff)
function DTHETA=risoluzione_THETA (t,THETA,ROT,Wb_eff)
```

Restituiscono l'orientamento del drone espressa in O_E .

Essendo una ripetizione di integrazioni in piccoli istanti è stato necessario inserirle in un ciclo “for” dove venivano scansionati tutti gli intervalli.

Saranno presenti, perciò, una serie di valori dipendenti dal tempo in ogni intervallo ottenuto già in precedenza.

I valori quindi in totale disponibili nel tempo di simulazione questa volta saranno molto maggiori.

Saranno presenti tante iterazioni tanti quanti saranno gli intervalli temporali utilizzati per il calcolo delle grandezze in assi corpo con la serie precedente di “ode45”.

La prima iterazione, tuttavia, viene fatta all'esterno del ciclo “for” in quanto il punto iniziale per l'integrazione è dato come input tra le condizioni iniziali.

Vengono introdotte infatti x, y e z in P_e e ϕ, θ e ψ in A_e , come prima in minuscolo

Per tutte le altre iterazioni, viene preso come punto iniziale l'ultimo punto integrato nell'iterazione precedente così da ottenere una continuità nei valori.

I valori risultanti rappresentano la variazione dei vettori posizione nei tre assi, in X, Y e Z e orientamento, in ϕ, θ e ψ .

L'integrazione numerica di tanti segmenti restituisce una serie di curve di secondo grado che si susseguono. Se gli intervalli sono sufficientemente piccoli la curva risultante ad occhio nudo somiglierà ad un'unica grande parabola.

Graficando tramite la funzione “plot” l'evolversi di traslazione e angoli di Eulero si ottengono i dati desiderati: ovvero note le forze e i momenti agenti sul mezzo, il quadricottero effettuerà movimenti o manterrà una direzione desiderata con eventuali variazioni di assetto.

2.2 Implementazione a Blocchi Simulink:

L'implementazione seguente svolta in Simulink serve a mostrare che gli stessi risultati sono ottenibili anche con questo strumento.

Lo scopo resta quello di una simulazione semplificata per valutare la fattibilità di manovre semplici atte a testimoniare la validità del modello implementato.

Vista l'ingente quantità di dati si è scelto di dividere l'intero modello in blocchi indipendenti tramite il blocco "subsystem".

Al loro interno sono stati svolti tutti i calcoli numerici parziali a noi necessari.

Tutti i dati necessari riguardo al quadricottero sono introdotti tramite uno script in Matlab e vengono trasferiti dal Workspace, la finestra dove sono elencate le variabili presenti, allo schema Simulink tramite l'assegnazione con il blocchetto "constant".

In questo modo passano dati strutturali del velivolo, regime di rotazione dei rotori, condizioni iniziali di grandezze assi corpo (V_b e W_b) ed i singoli elementi delle matrici di rotazione ROT e DCM.

Il primo calcolo necessario è stato quello delle forze e dei momenti esterni agenti.

Per effettuare questa operazione si sono utilizzati due sottosistemi uno dentro l'altro:

- "spinta&momenti" per il calcolo delle forze di trazione dei motori e dei momenti in funzione della velocità dei rotori;
- "forze&momenti" assemblato per il calcolo finale delle forze nelle sue tre componenti con l'utilizzo della matrice DCM.

Successivamente a questa operazione siamo in grado di risolvere le equazioni di Newton ed Eulero affinché si abbiano a disposizione le grandezze integrate nell'intervallo di simulazione in Oabc.

In "Risoluzione V_b " sono introdotte le tre equazioni scalari di Newton che permettono di trovare rispettivamente le grandezze u, v, w e tramite tre integratori posti a valle si è riusciti ad ottenere un andamento veritiero dei valori in gioco nel tempo.

In Matlab rappresentano i vettori tempo varianti che nell'implementazione dopo "ode45" erano chiamati U, V e W e vengono definiti per comodità così anche in Simulink.

Le condizioni iniziali devono essere introdotte anche nella finestra di setting dell'integratore stesso.

Se così non fosse l'andamento e la pendenza sarebbero corretti ma non sarebbe presente la condizione iniziale al tempo zero nel grafico.

Il sottosistema affine, "Risoluzione W_b ", risolve allo stesso modo le tre equazioni di Eulero restituendo in uscita le tre grandezze integrate di p, q, r .

Parimenti, questi sono equivalenti i vettori di Matlab P,Q,R e sono anche questi chiamati allo stesso modo. In due blocchi “scope” sono stati quindi graficati i risultati di questa prima integrazione.

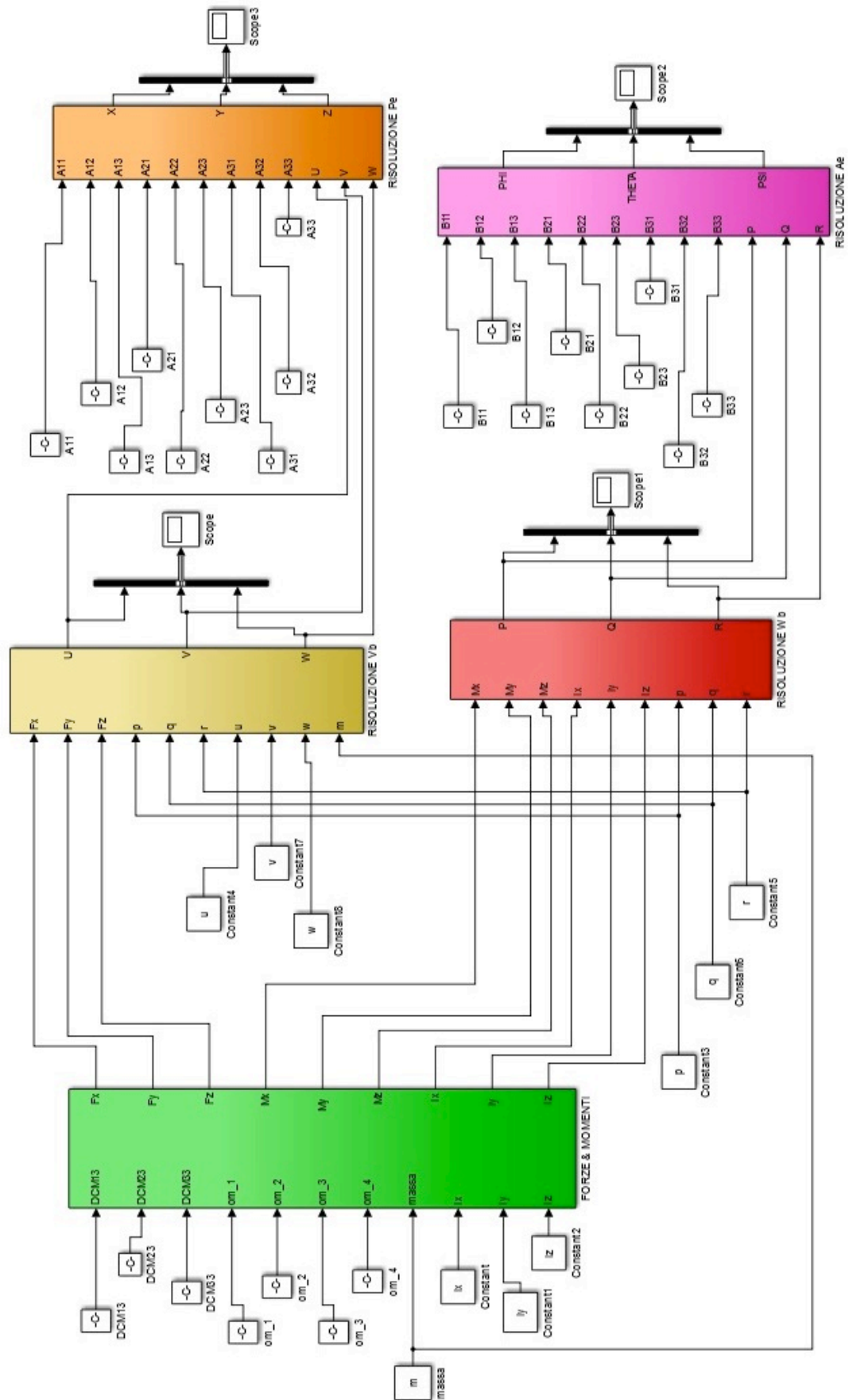
Appare chiaro che nella seconda metà del processo, si intende quantificare rispetto al tempo le grandezze nel sistema di riferimento in assi terrestri O_E .

I dati di cui si ha bisogno sono oltre ai vettori velocità di traslazione e rotazione sono le matrici inverse di DCM e ROT rinominate nel calcolo A e B.

L’operazione di inversione viene effettuata a monte nello script di partenza implementato in Matlab e la chiamata viene effettuata in Simulink settando opportunamente ciascun blocco “constant” come prima. Il calcolo vero e proprio viene svolto all’interno dei sottosistemi “Risoluzione P_e ” e “Risoluzione A_e ”. Il primo permette di trovare la soluzione integrata delle posizioni espressa in X,Y,Z mentre il secondo ottiene dati sugli sfasamenti angolari degli angoli di Eulero integrati in PHI, THETA, PSI. Come nei sottosistemi precedenti l’operazione è stata eseguita tramite integratori posti a valle delle singole equazioni scalari. Le condizioni iniziali sono poi state eventualmente inserite all’interno a seconda delle singoli casi studiati.

Seguono come dato finale e di riscontro gli andamenti temporali delle ultime grandezze trovate a partire da forze e momenti ottenuti e dai dati di assetto di partenza del drone.

Segue alla pagina seguente lo schema globale completo di tutti i sottosistemi utilizzato in Simulink.



2.3 Manovra di Hovering:

Il volo a punto fisso, oppure hover, rappresenta lo stazionamento in volo a velocità nulla di un velivolo. Come prima citato, il bilancio delle forze è dato semplicemente da:

- forza peso perpendicolare al piano appartenente al quadricottero diretta verso il basso;
- spinta data dai quattro motori diretta allo stesso modo della forza peso ma verso l'alto.

Non dimentichiamo che i motori sono controrotanti a due a due: questo significa che la spinta dovuta alla rotazione delle pale delle eliche può, oltre ad eguagliare il peso, generare momenti intorno agli assi di rotazione del mezzo. Per l'annullamento delle coppie occorrerà rendere nulli i termini u_2, u_3, u_4 . L'unica opzione è far girare i motori alla stessa velocità di rotazione.

Nella simulazione, di 10s, si otterrà dalla posizione di partenza, uno stazionamento così descritto.

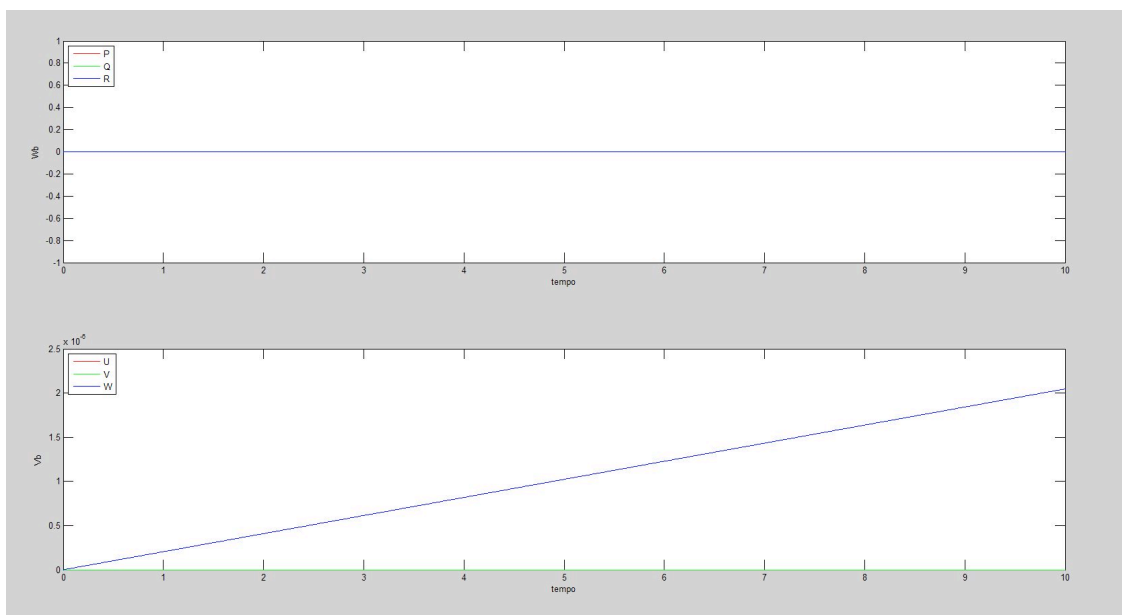
Questo rappresenta il modello più semplice da simulare. Tutte le condizioni iniziali quali sono angoli di assetto, velocità di rotazione e traslazione sono nulle.

2.3.1 Modello Matlab Hovering:

A parte i parametri fisici e geometrici del quadricottero che sono fissi si sono immesse le condizioni iniziali qui rappresentate e calcolato con il bilancio di forze sopra citato il giusto valore di spinta:

$$\begin{bmatrix} u, v, w \end{bmatrix}^T = 0 \quad \begin{bmatrix} x, y, z \end{bmatrix}^T = 0 \quad \begin{bmatrix} p, q, r \end{bmatrix}^T = 0 \quad \begin{bmatrix} \phi, \theta, \psi \end{bmatrix}^T = 0$$

La serie di grafici che segue rappresenta i risultati:



Fig,12 : Nel grafico i risultati in assi corpo, W_b e V_b .

L'unico lievissimo spostamento nell'asse verticale è dato dal fatto che la soluzione matematica al problema è nella pratica impossibile da raggiungere: il valore del numero dei giri dei rotori non può essere infinitamente preciso per ottenere una retta dritta come le altre.

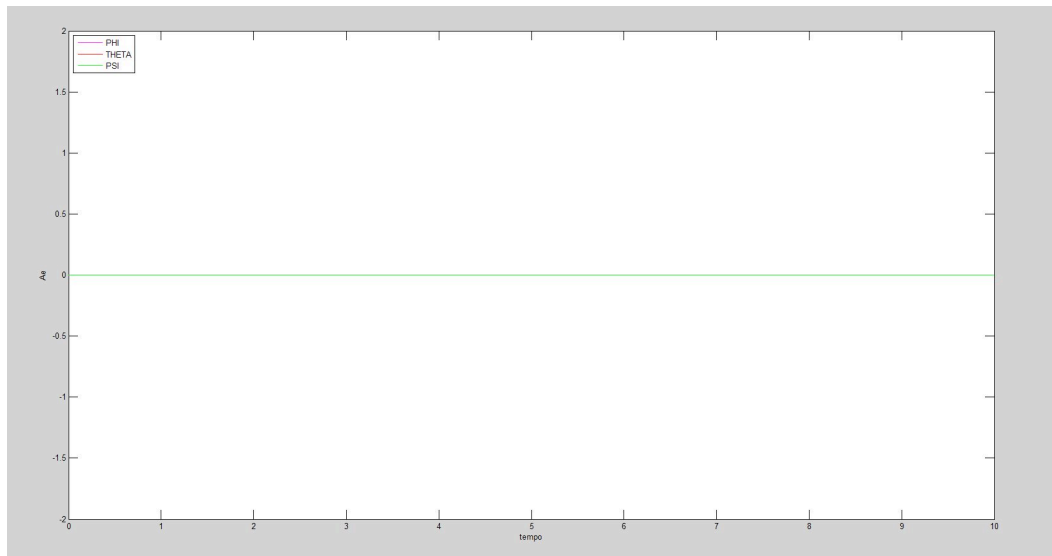


Fig. 13: Angoli di Eulero in hovering, A_e .

Per quanto riguarda l'assetto complessivo era auspicabile che questo restasse totalmente stazionario.

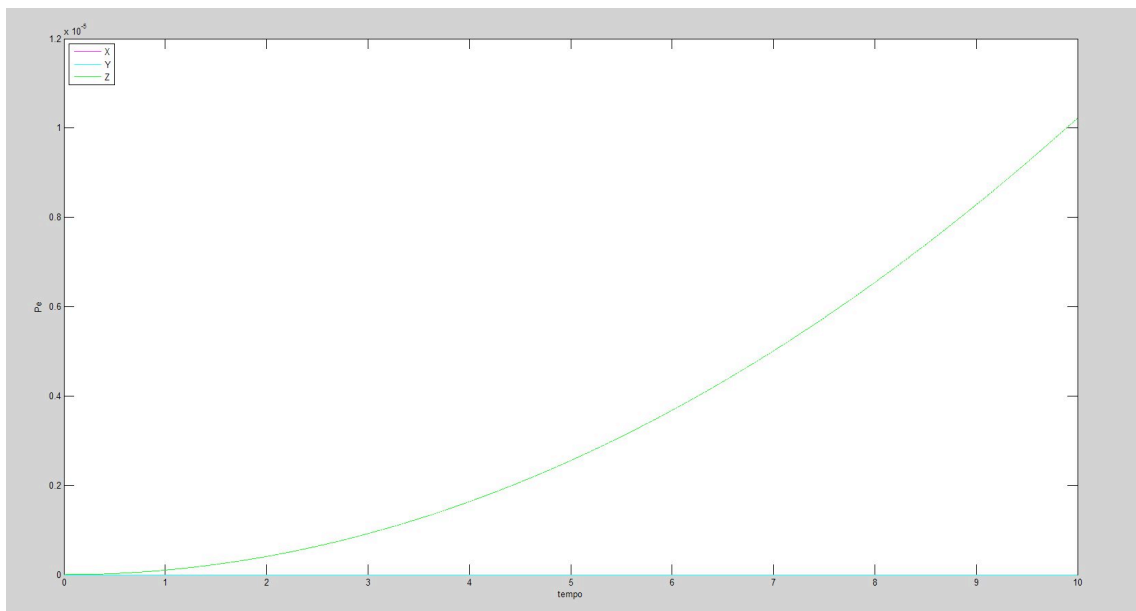


Fig.14: Spostamenti in assi fissi terrestri, P_e .

Lo spostamento dato dalla terza componente di P_e è una diretta conseguenza della velocità verticale di V_b ottenuta nel grafico precedente.

2.3.2 Modello Simulink Hovering:

A parità di modello implementato, quello che ci si aspetta è di trovare gli stessi grafici ottenuti con la scrittura in Matlab. Li riportiamo, quindi, testimoniando l'uguaglianza dei risultati:

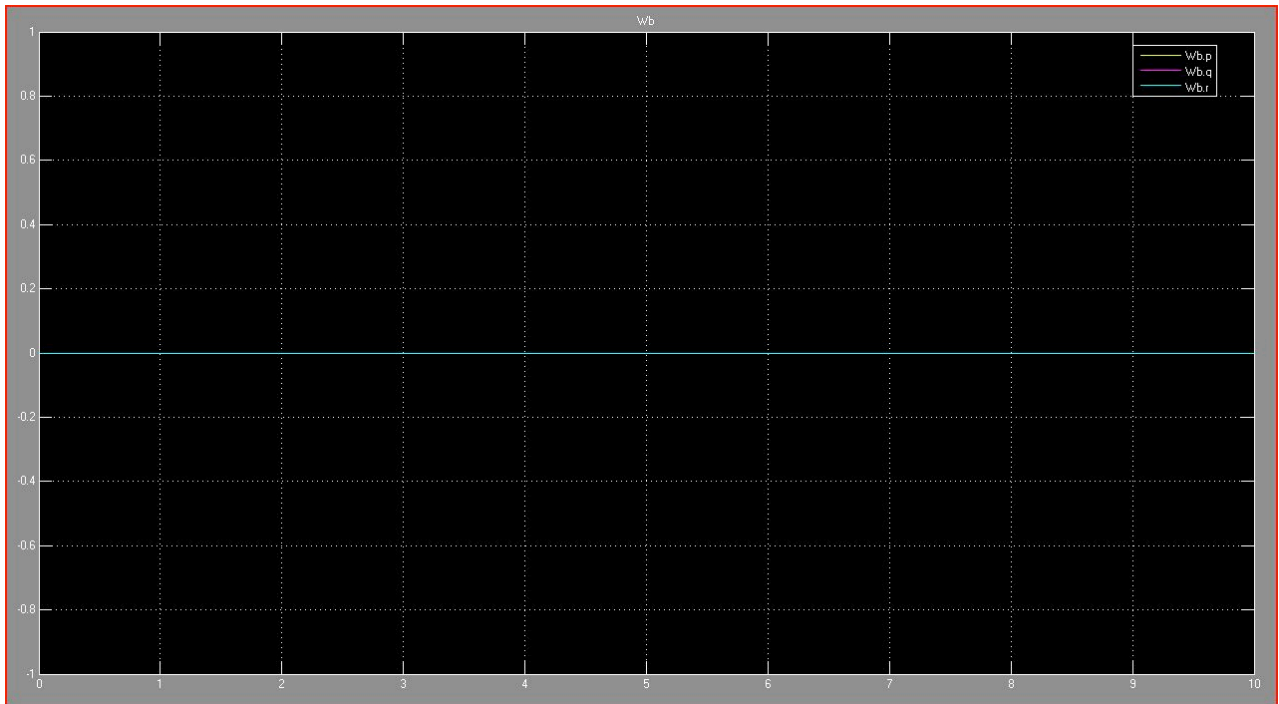


Fig.15: Plot in Simulink di W_b .

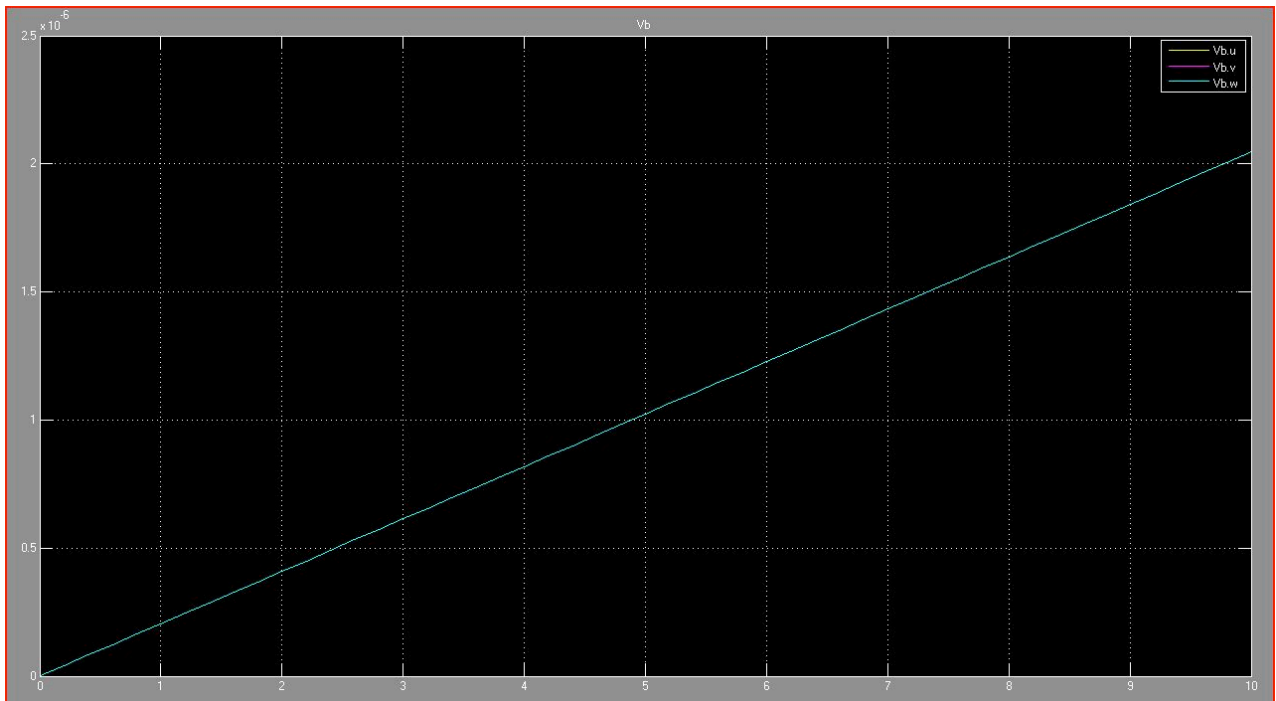


Fig.16: Rappresentazione di V_b .

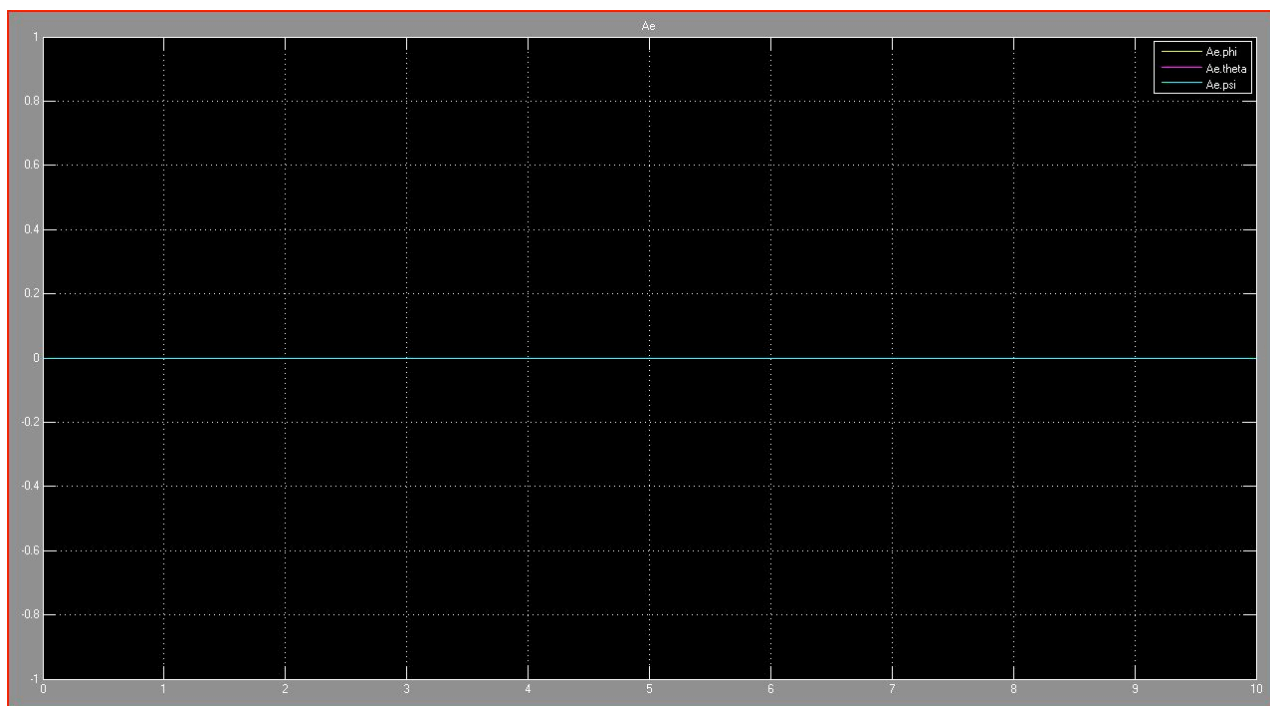


Fig.17: Assetto in Hovering, A_e .

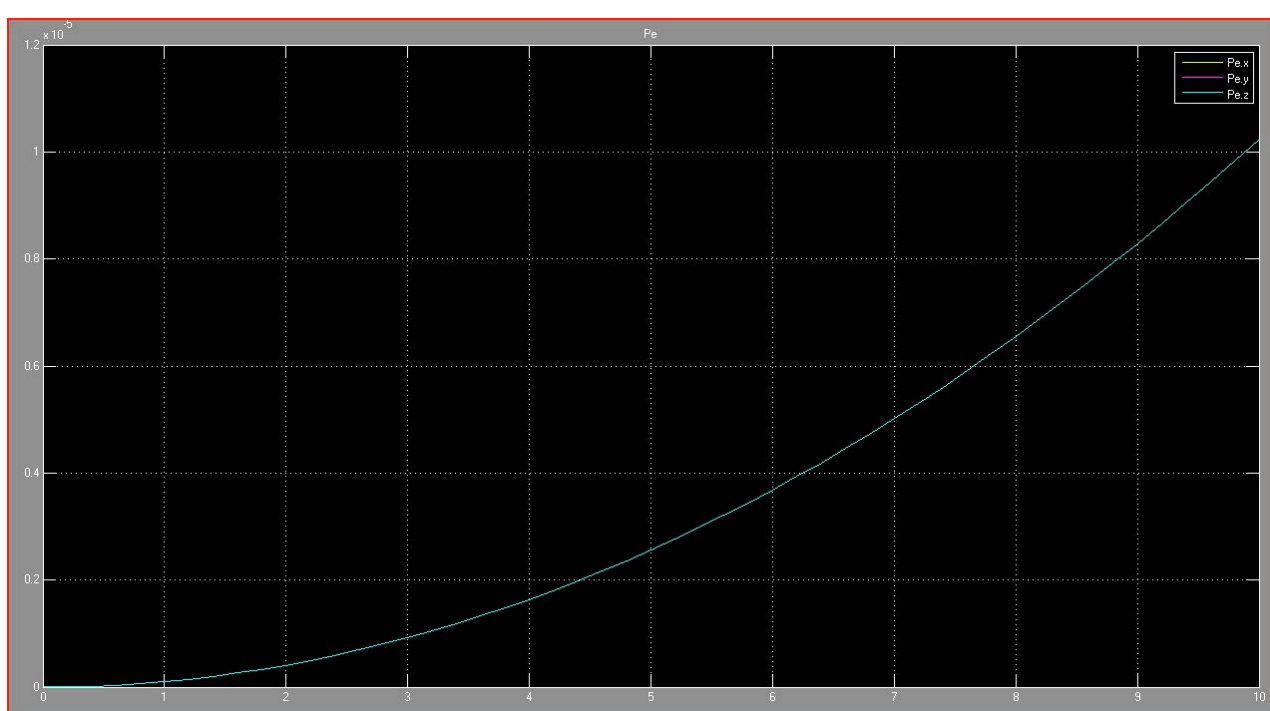


Fig.18: Lo stesso spostamento in verticale espresso nella terna terrestre, P_e .

2.4 Manovra di Salita:

In generale i multicotteri per motivi costruttivi possiedono capacità VTOL (Vertical Take Off and Landing): la spinta iniziale da fermo è sempre verticale.

La manovra di salita semplice non prevede inoltre variazioni di assetto.

Come per l'hovering quindi la forza di riferimento resta la forza peso: questa volta però non è sufficiente eguagliarla ma superarla. Di quanto dipende soltanto dall'utente o comunque da quanto si vuole rapida questa manovra. Nella simulazione è stata voluta una accelerazione di 1m/s^2 per una durata di 10s. Anche in questo caso, le coppie dovranno annullarsi per non generare momenti angolari: tutti i propulsori forniranno perciò la stessa spinta. Le eliche però ruoteranno a valori superiori per garantire il surplus di potenza.

2.4.1 Modello Matlab di Salita:

Le condizioni iniziali riguardanti la descrizione di una traiettoria rettilinea verticale poggia sulle stesse condizioni iniziali di quelle della simulazione precedente:

$$\begin{bmatrix} u, v, w \end{bmatrix}^T = 0 \quad \begin{bmatrix} x, y, z \end{bmatrix}^T = 0 \quad \begin{bmatrix} p, q, r \end{bmatrix}^T = 0 \quad \begin{bmatrix} \phi, \theta, \psi \end{bmatrix}^T = 0$$

L'unica modifica va eseguito un semplice calcolo della forza necessaria per ottenere l'accelerazione desiderata: si ha così un nuovo valore di regime dei motori.

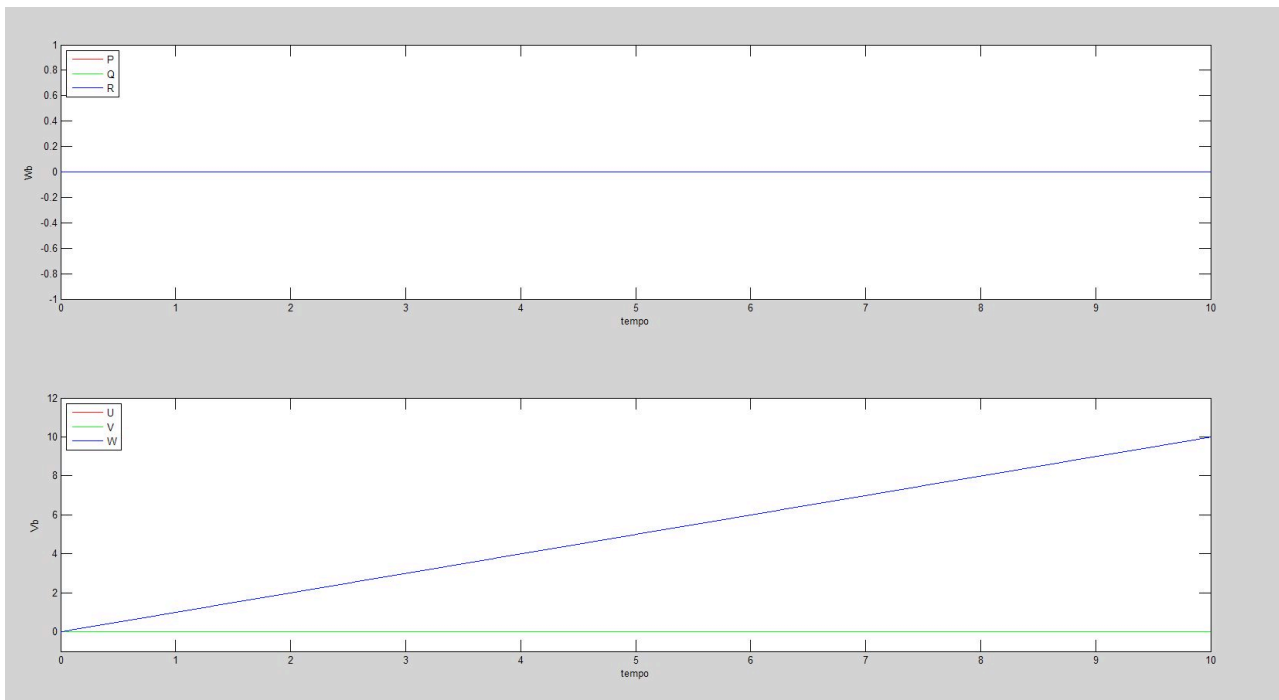


Fig.19: L'aumento di velocità aspettato nella componente verticale in V_b .

Essendo la forza verticale ci si aspetta che le variazioni siano soltanto nell'ultima componente della velocità traslazionale.

Essendo una salita semplice non si prevederanno invece variazioni di assetto dovute a velocità rotazionali espresse in assi corpo.

Lo spostamento che ne risulterà sarà sulla scia, come in precedenza, della velocità ottenuta.

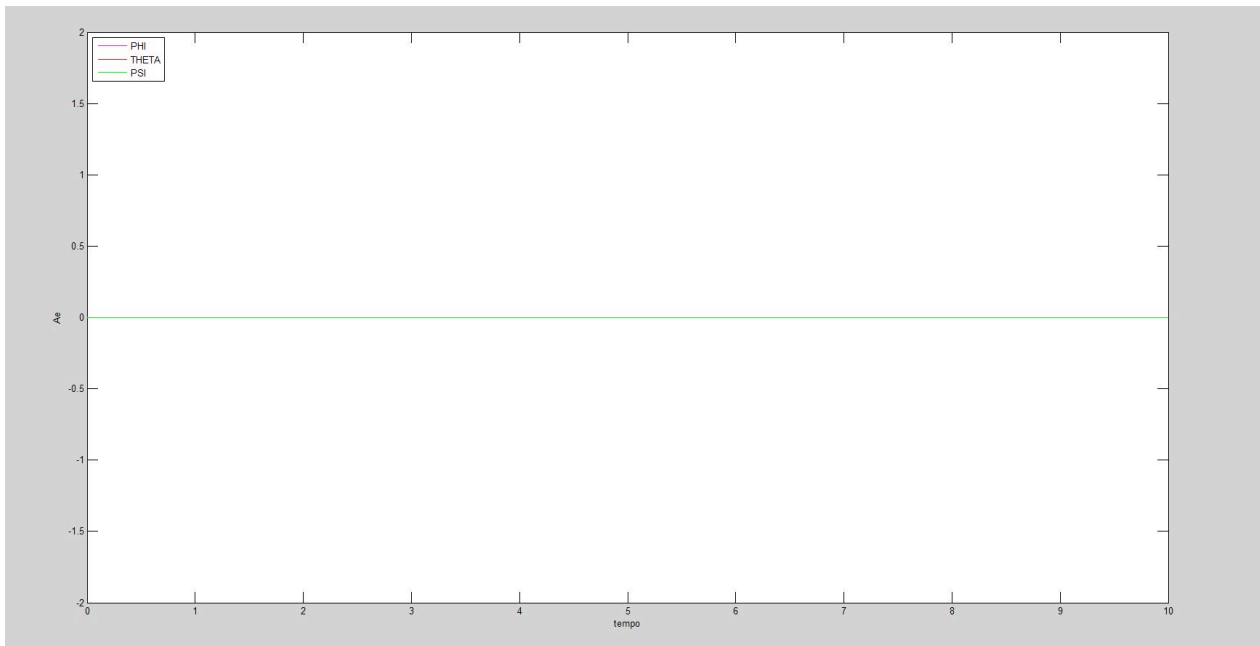


Fig.20: Assetto costante piano di salita mostrato in A_e .

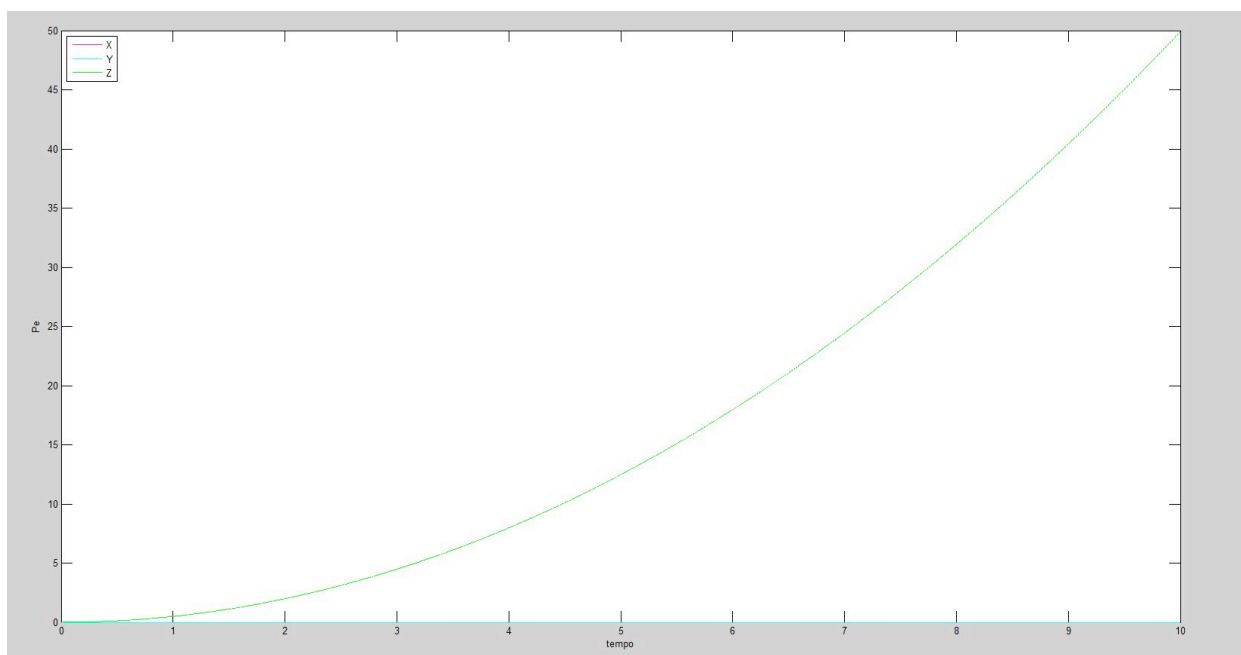


Fig.21: Lo spostamento ottenuto, P_e .

2.4.2 Modello Simulink di Salita:

Con il modello implementato in Simulink si è replicato il risultato ottenuto che mostriamo:

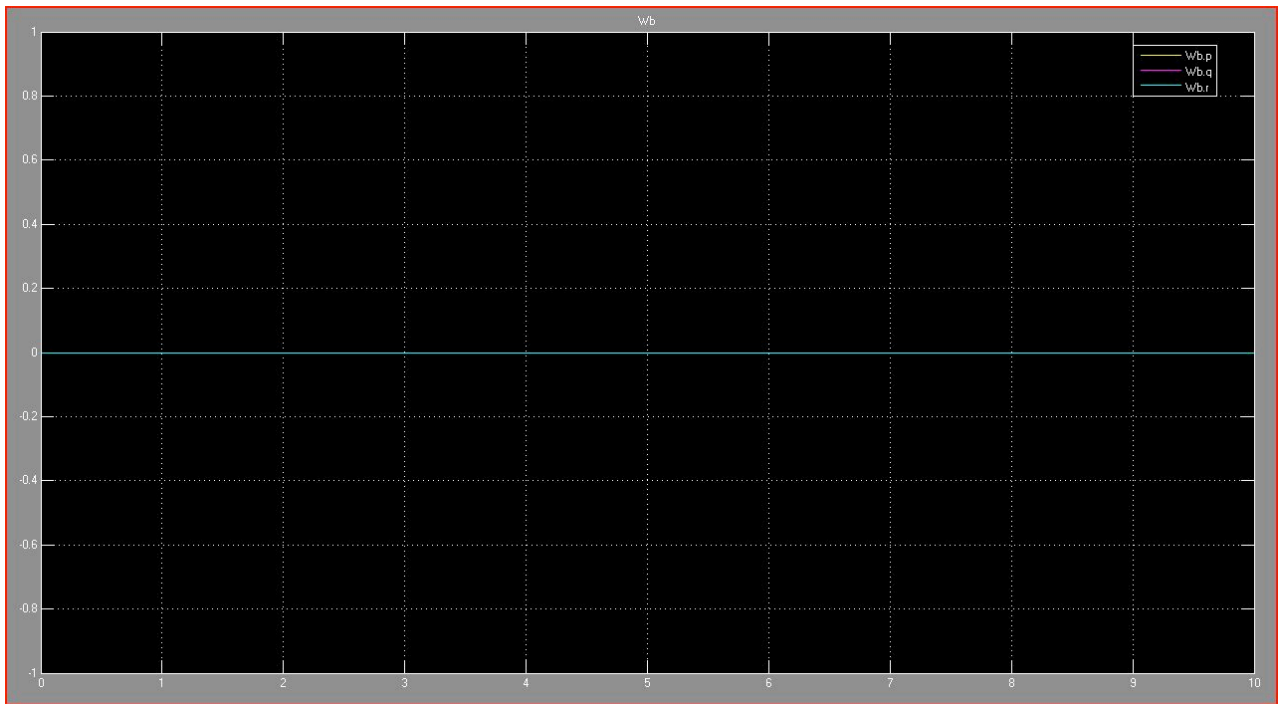


Fig.22: Rotazioni nulle in assi corpo, W_b .

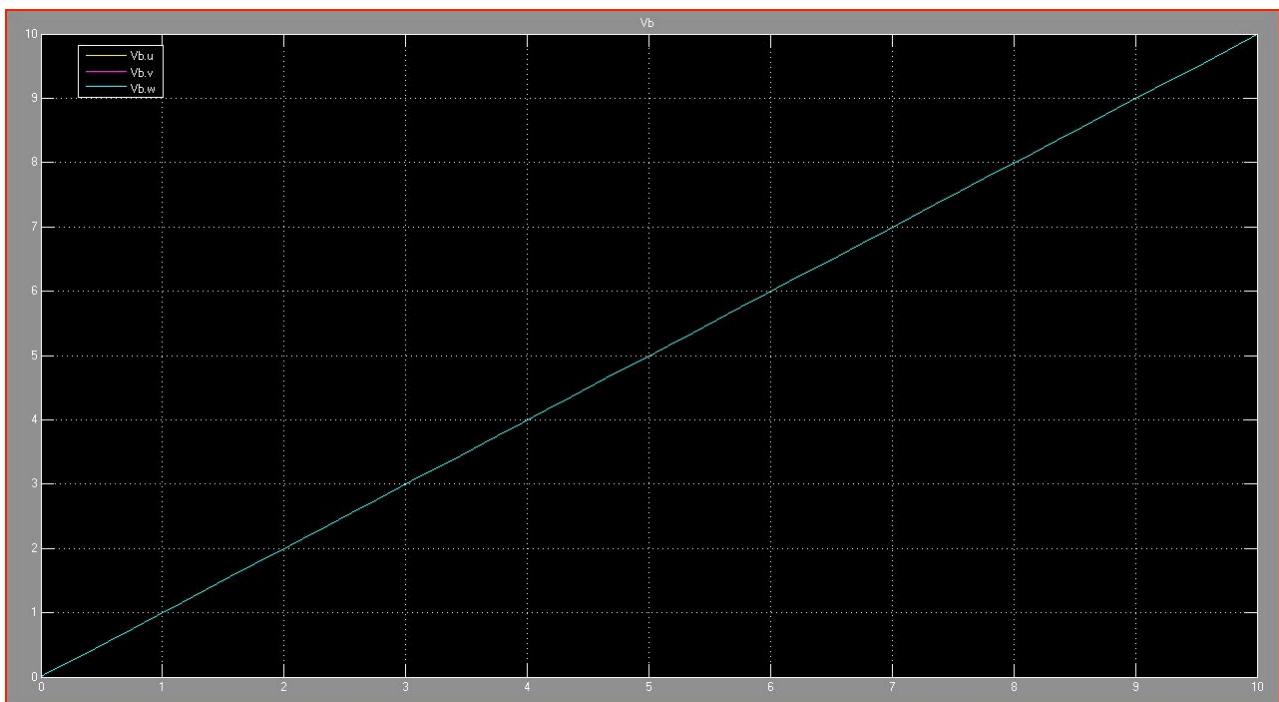


Fig.23: Rappresentazione di V_b .

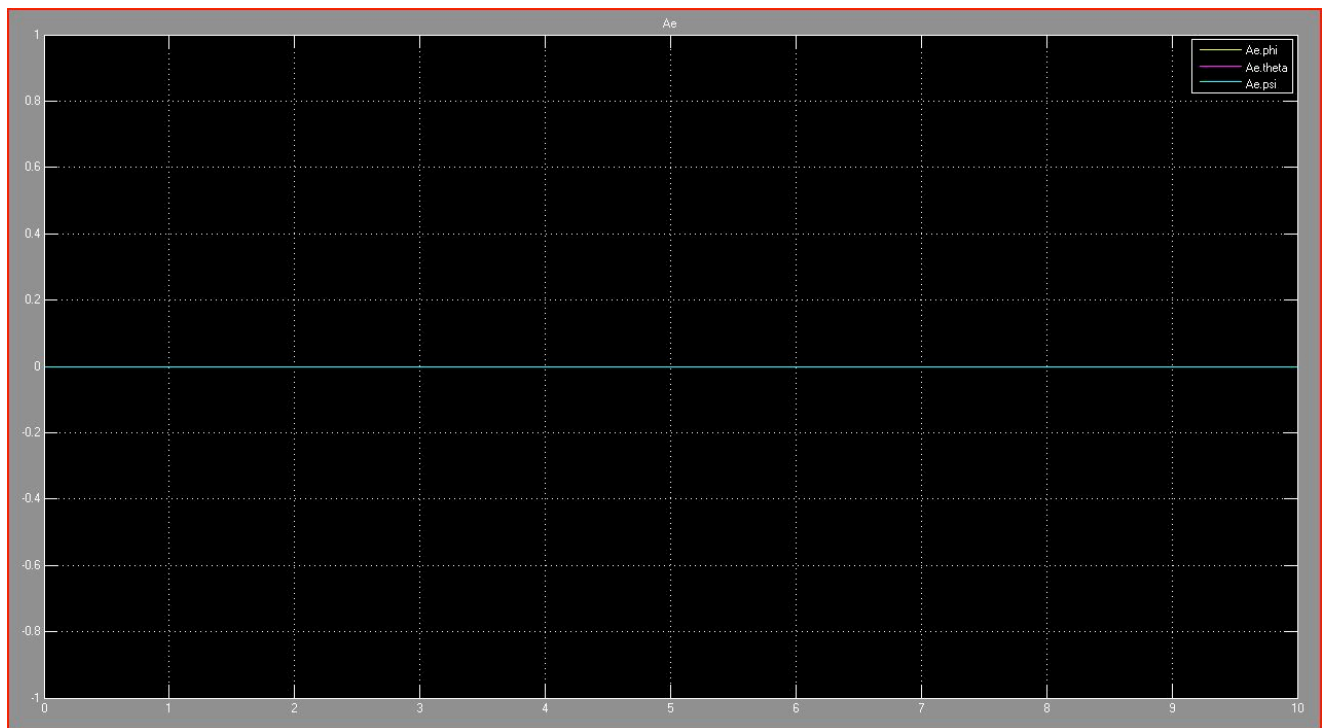


Fig.24: L'Assetto mantenuto costante in A_e .

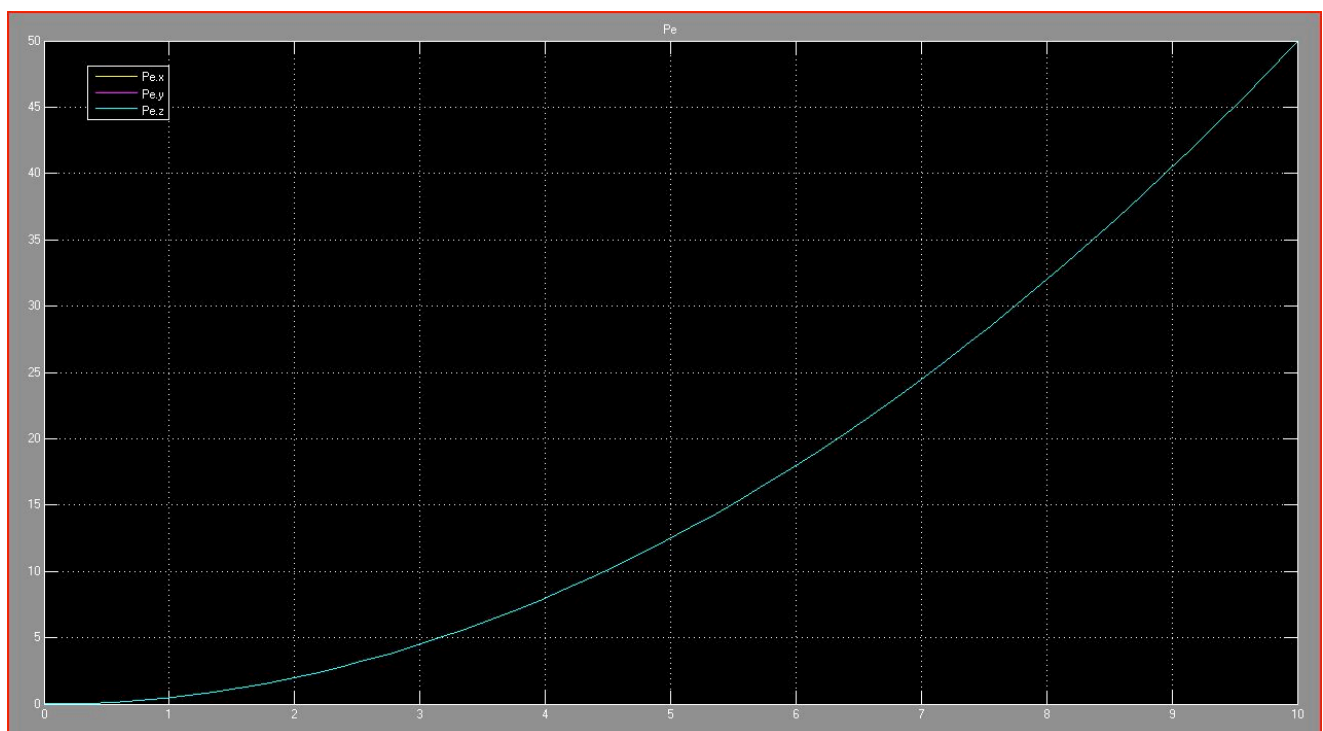


Fig.25: Lo stesso spostamento verticale ottenuto, P_e .

2.5 Manovra di Volo Orizzontale:

Per ottenere un volo traslato è necessario che sia presente una componente residua di forza nella direzione del moto che permetta questa manovra.

La spinta come oramai sappiamo è perpendicolare al piano del quadricottero e l'unico modo che si ha per avere una componente orizzontale è inclinare il drone attorno all'asse di pitch (θ).

L'angolo di pitch è considerato negativo se il drone abbassa il muso.

Come conseguenza di ciò, i motori forniscono una spinta che non è verticale: si avranno questa volta due componenti, una verticale ed una anche orizzontale.

Quella verticale dovrà pareggiarsi con la forza peso mentre l'altra dovrà produrre il movimento, in avanti se il pitch è negativo.

Nella simulazione lo stato iniziale del quadricottero si presenta a manovra già avviata, quindi non si vede il transitorio di passaggio dallo stato orizzontale allo stato inclinato e quindi non parte da fermo. Si suppongono un'inclinazione di -10° , con velocità di avanzamento espressa in assi corpo nelle condizioni iniziali.

In questo modo non è sufficiente definire una componente sola del vettore V_b , causa inclinazione, bensì due. La traslazione sarà una somma vettoriale della componente "u" e "w".

Ovviamente la presenza di un angolo ben definito, definirà automaticamente il rapporto tra i valori "u" e "w".

Le forze dovranno mantenere il quadricottero all'angolo di assetto prescelto, senza nessuna variazione di coppia quindi.

Siccome noi non rappresentiamo il transitorio, anche in questo caso i giri dei vari rotori avranno gli stessi valori, sicuramente superiori a quelli di hovering, poiché qui il peso si scarica su una sola componente della forza di spinta.

2.5.1 Modello Matlab di Manovra Orizzontale:

Come sarà chiaro, ci si appresta anche questa volta le condizioni iniziali descriventi il moto del nostro quadricottero al tempo zero della simulazione:

$$[x, y, z]^T = 0$$

$$[p, q, r]^T = 0$$

$$[u, v, w]^T = [2, 0, 0.352]^T$$

$$[\phi, \theta, \psi]^T = [0, -10, 0]^T$$

La velocità di partenza a manovra già effettuata è supposta bassa e assunta a 2 m/s lungo la direzione frontale in O_{ABC} , che non rappresenta vista l'inclinazione del mezzo la direzione rettilinea come la immaginiamo.

Il valore di 0.352 m/s è stato estrapolato calcolando il valore necessario ottenendolo dal seno dell'angolo impostato per la traiettoria così da avere una direzione rettilinea in assi fissi terrestri.

Anche durante l'accelerazione progressiva si nota che il rapporto tra la prima e la terza componente di V_b è sempre costante proprio a mantenere l'assetto fisso.

Quello che si è ipotizzato è stato di fatto riscontrato numericamente:

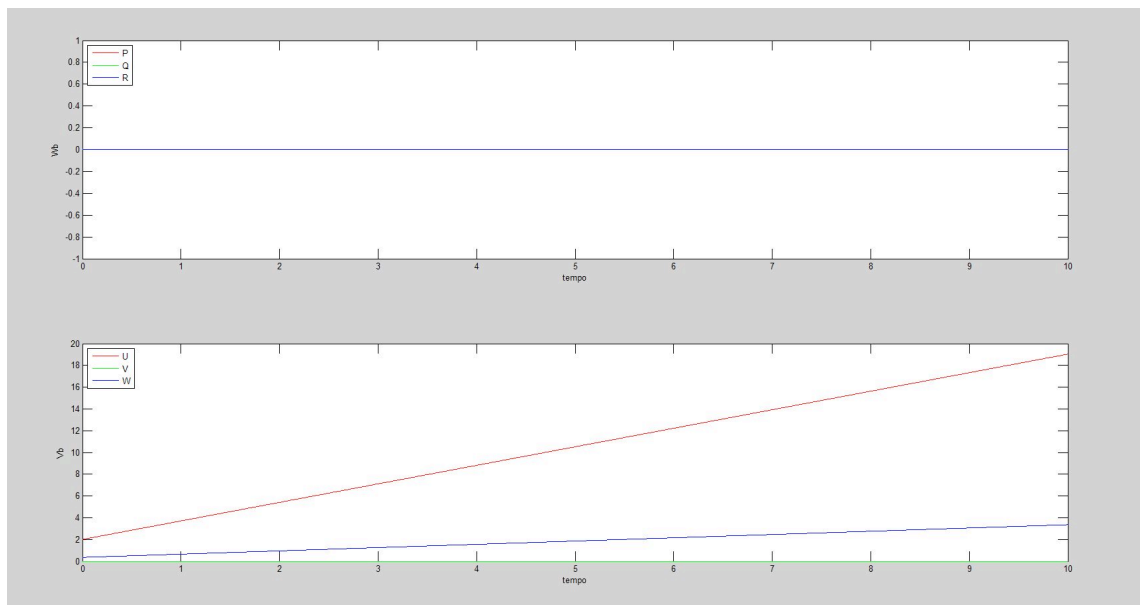


Fig.26: Si evincano le due componenti di V_b nel grafico inferiore.

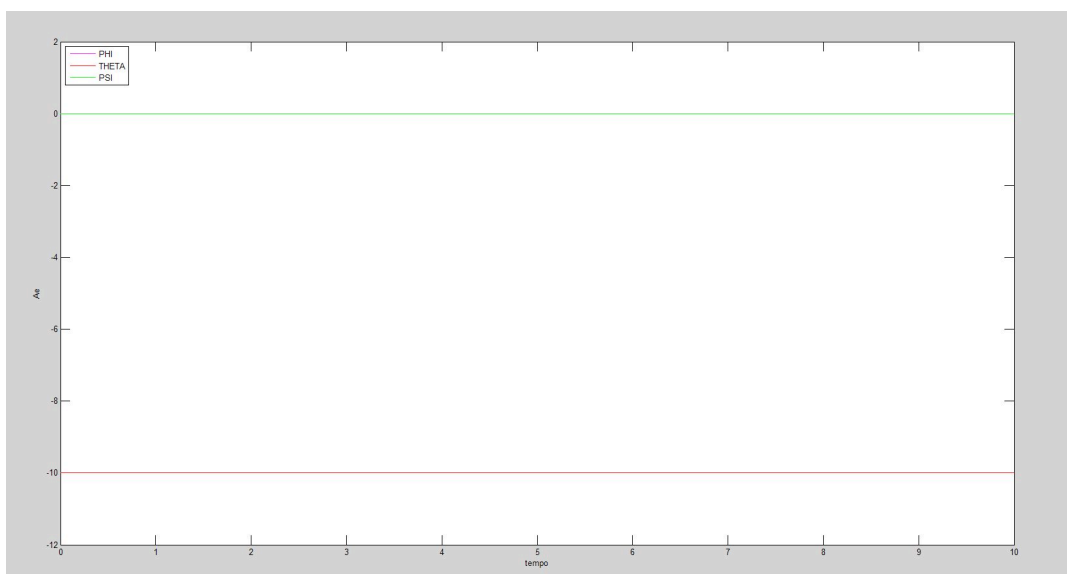


Fig.27: Gli angoli di A_e che restano costanti.

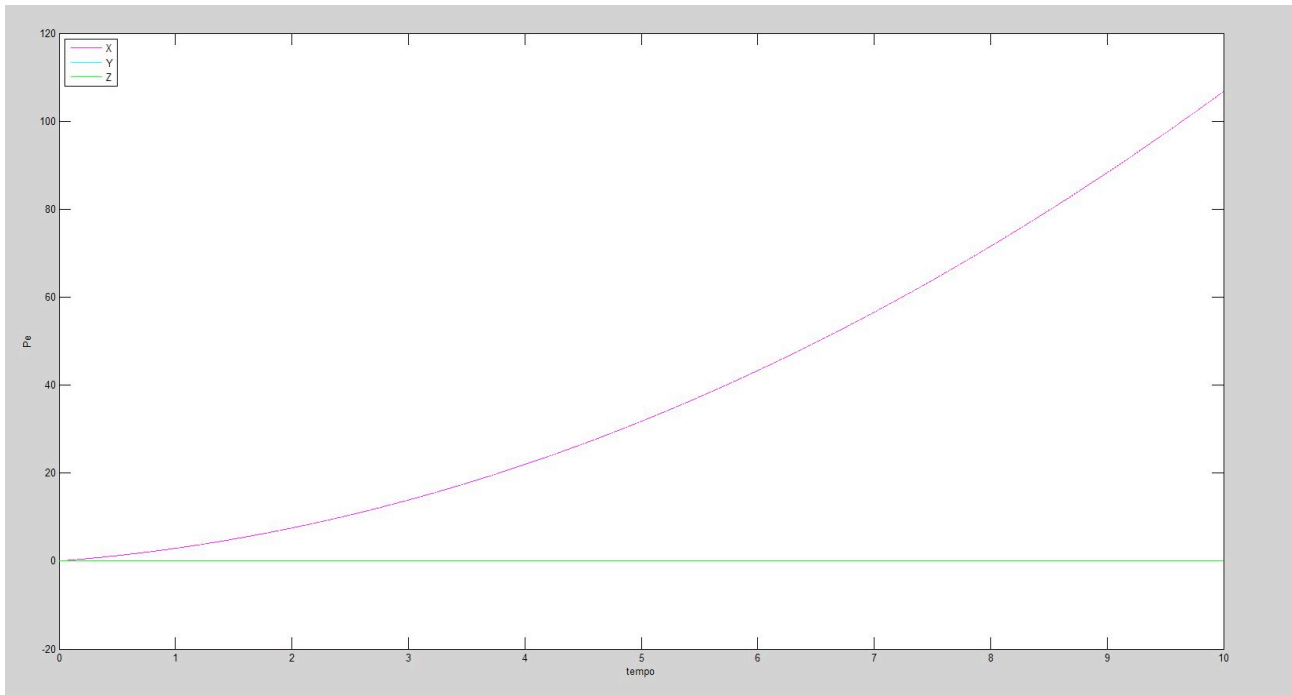


Fig.28: Lo spostamento finale da notare questa volta in una sola direzione in assi terrestri.

2.5.2 Modello Simulink Manovra Orizzontale:

Anche in quest'ultimo caso si è visto come i modelli coincidano come modellazione e simulazione del veicolo. I plot osservati:

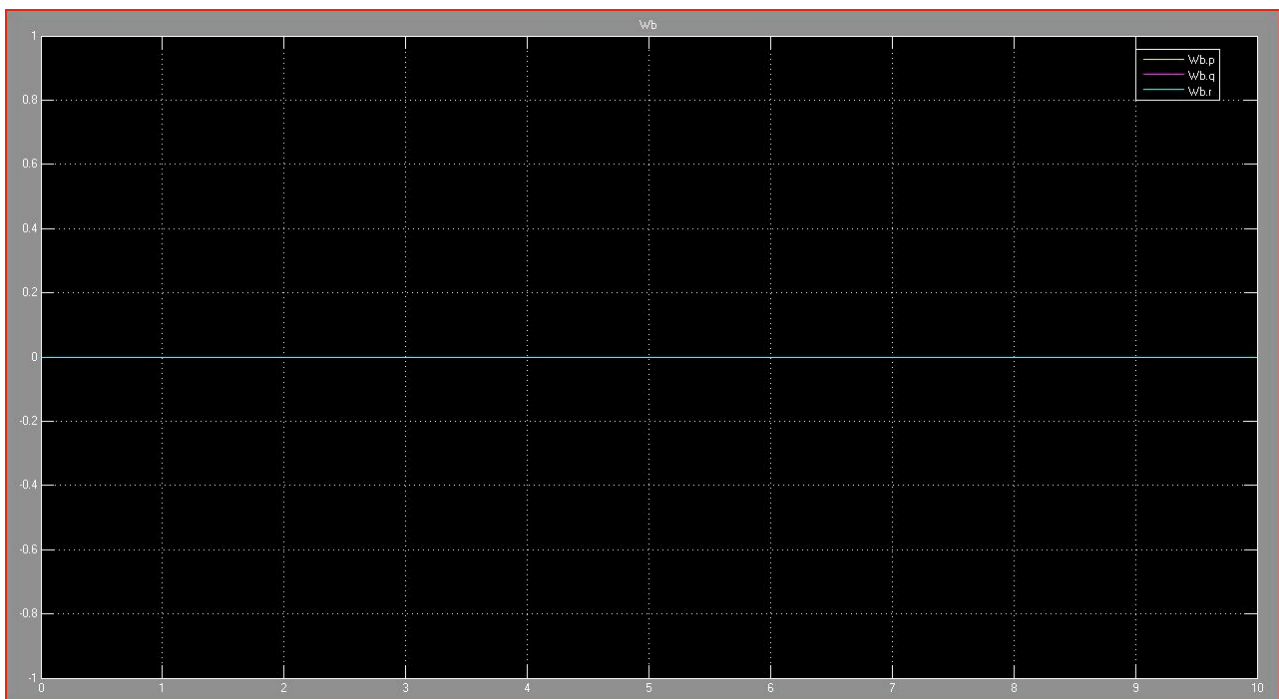


Fig.29: Il riscontro di W_b in Simulink.

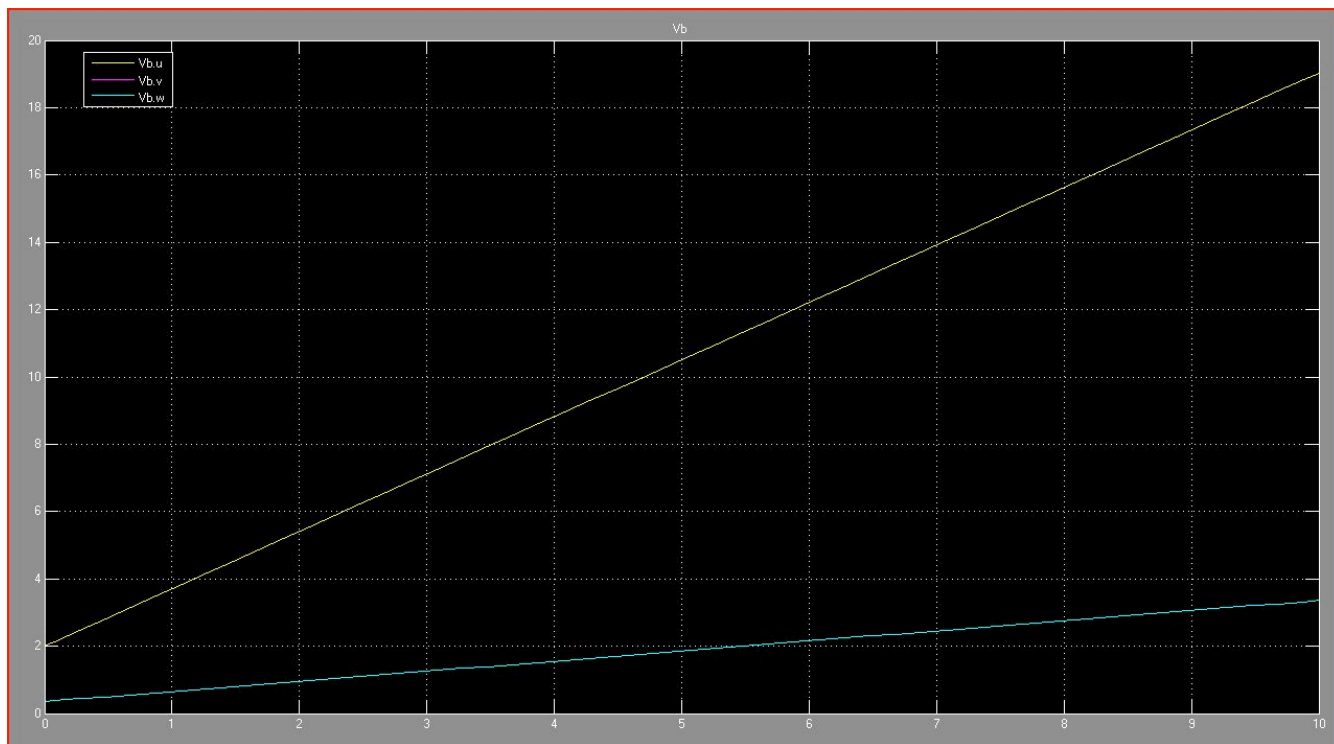


Fig.30: Velocità di traslazione nel multirotore, V_b .



Fig.31: Assetto di avanzamento, A_e .

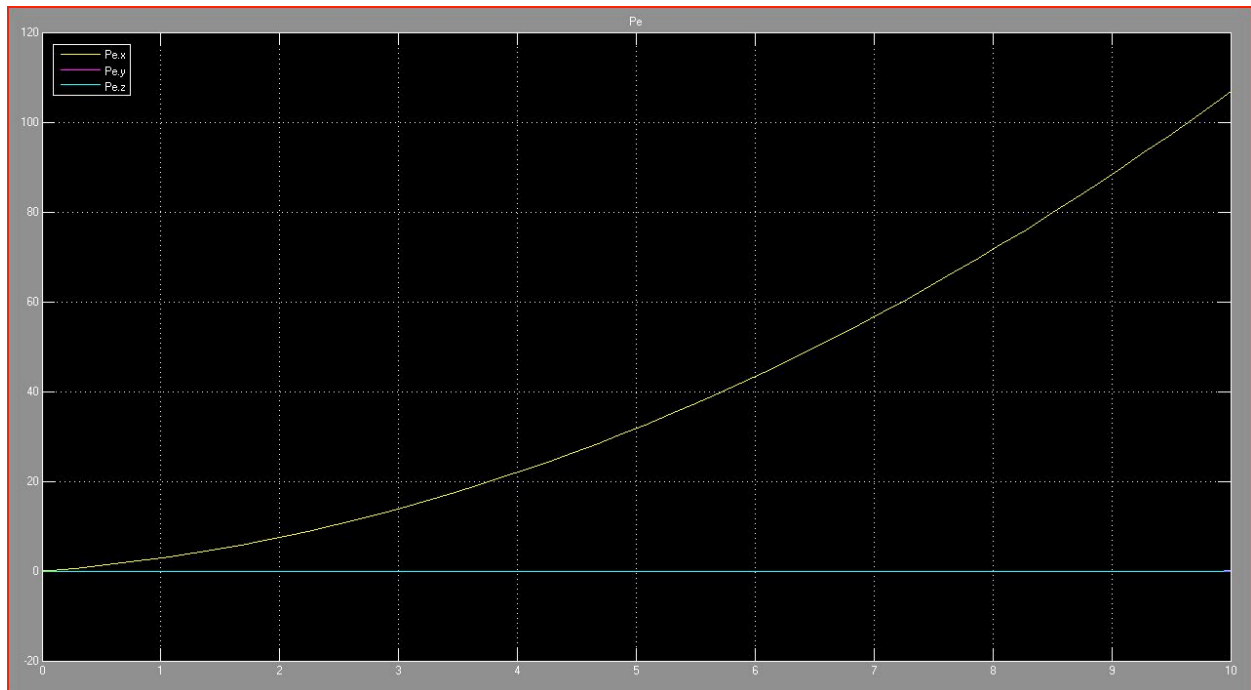


Fig.32: Lo spazio percorso dal velivolo in O_E .

2.6 Confronto dei Modelli e Tempistiche:

Avendo possibilità di confronto si scopre che l'integrazione in Matlab è notevolmente influenzabile dal numero di intervalli temporali considerati per la simulazione. Inoltre tramite l'utilizzo in ambiente Matlab di "ode45" in serie costringe l'utente ad aumentare progressivamente il numero di istanti considerati, avendo il bisogno di integrare in intervalli precedentemente integrati. Ciò ovviamente comporta ad un aggravio di tempo e di costo computazionale per avere dei risultati piuttosto precisi. Per quanto concerne Simulink, invece, si nota da subito maggior rapidità nel graficare le nostre soluzioni. In più al variare del metodo di integrazione, a passo fisso o variabile, o del metodo di risoluzione (ode45 per passo variabile ed ode3,ode4,ode5 per passo fisso) i risultati sono gli stessi. Quindi per la risoluzione di questo tipo di problemi è maggiormente intuitivo e veloce utilizzare Simulink. In particolare, portando all'estremo le potenzialità dello script in Matlab, nel dettaglio i valori ottenuti tendono a quelli in Simulink, dove invece sono sempre uguali: con l'utilizzo di un milione di valori in Matlab si ottengono valori ancora differenti, seppur nell'ordine dello 0,1%, da quelli ottenibili in Simulink con meno di sessanta valori con integrazione a step variabile. Seppur di default anche Simulink utilizzi "ode45", non possiede lo stesso funzionamento dell'omonimo risolutore di Matlab. Nel primo il risolutore è un codice built-in compilato mentre nel secondo è un file scritto (.m file). Inoltre i valori di default dei parametri risolutivi propri del metodo sono leggermente differenti. L'utilizzo dello schema a blocchi inoltre permette la computazione più velocemente. Tutto questo indica un vantaggio nell'uso intuitivo di Simulink.

Cap.3: Fisica di un Quadricottero con Payload Sospeso

La seconda parte del lavoro svolto è stato svolto ponendo l'attenzione su un problema simile, ma di maggior complessità: si è scelto di descrivere il moto del nostro quadricottero a cui è stato aggiunto però un payload sospeso. Il moto che ne risulta quindi è un sistema massa-pendolo dove la massa è rappresentata dal peso del veicolo.

La complessità maggiore risiede nel fatto che bisogna introdurre legami tra il movimento oscillatorio del carico e l'assetto complessivo del quadricottero.

Per questo bisogna rivedere il modello matematico e cambiare approccio per la ricerca di una possibile soluzione. Nella modellazione precedente si sono utilizzate le equazioni di Eulero-Newton per la risoluzione, tuttavia, una possibilità è rappresentata anche dalle equazioni di Lagrange.

3.1 Ambiente Matematico:

Prima di spiegare nel dettaglio cosa le equazioni lagrangiane rappresentino, dobbiamo fare delle assunzioni semplificative a livello fisico. Il legame tra payload e velivolo è realizzato tramite un cavo: anch'esso deve essere ben definito. In particolare, si assume che il cavo sia agganciato alla proiezione del baricentro giacente sul piano più basso appartenente al drone.

Inoltre, le proprietà di massa del cavo non sono tenute in considerazione ed è considerato inestensibile. Se così non fosse, si sarebbero dovuti tenere in considerazione gli effetti, anche transitori, elastici del cavo. Il peso invece è modellato come un elemento puntiforme.

La descrizione matematica di questo sistema più ampio non può quindi essere effettuata utilizzando i classici sei gradi di libertà già introdotti.

A questi sei che rappresentano lo spostamento (x,y,z) e la rotazione (φ, θ, ψ) nei tre assi fissi terrestri dobbiamo aggiungere altri due, che chiameremo λ e μ , per il carico trasportato.

Definiamo:

- λ : l'angolo presente tra il cavo e il versore locale verticale appartenente al piano su cui giace il versore rappresentante la direzione frontale;
- μ : l'angolo presente tra il cavo e il versore locale verticale appartenente al piano su cui giace il versore rappresentante la direzione laterale;

Anche se formalmente non saranno introdotti domini che rappresentano valori numerici precisi appare ragionevole che i limiti di posizionamento del carico, tenendo a mente inoltre la inestensibilità del cavo, non potrebbero superare i 90°.

3.2 Equazioni di Lagrange:

Come già introdotto le equazioni lagrangiane sono una seconda valida via per conoscere il moto di un sistema. Esse descrivono il moto di un oggetto che risponde al secondo principio della dinamica e mettono in relazione la posizione e la velocità di ogni elemento presente, in modo che sia possibile caratterizzarne completamente la dinamica. Posizione e velocità sono messe in correlazione tramite la manipolazione dei termini di energia cinetica e potenziale.

Primo step coincide con il determinare l'energia cinetica.

Questa può essere espressa mediante la somma di tre contributi principali:

- energia cinetica di traslazione del mezzo (1);
- energia cinetica di rotazione sempre del nostro drone (2);
- energia cinetica del carico (3).

$$K = \frac{1}{2} m V^2 + \frac{1}{2} \omega^T (I \omega) + \frac{1}{2} m_B V_B^2$$

Supposto che la lunghezza del cavo sia "l", a partire dalla posizione del baricentro in assi terrestri, che ricordiamo essere $r = (x, y, z)^T$, possiamo calcolare la posizione della massa sospesa grazie ai due nuovi gradi di libertà :

$$r_B = (x + l \cos \mu \sin \lambda, y + l \sin \mu, z + \cos \lambda \cos \mu)^T$$

Note le posizioni in assi terrestri, si calcola facilmente la velocità delle due masse eseguendo per ogni componente la derivata secondo la variabili presenti in ciascuna componente stessa:

$$V = (\dot{x}^2 + \dot{y}^2 + \dot{z}^2)^{\frac{1}{2}}$$

$$V_B = (\dot{x} + l(\dot{\lambda} \cos \mu \cos \lambda - \dot{\mu} \sin \mu \sin \lambda); \dot{y} + l \dot{\mu} \cos \mu; \dot{z} + l(\dot{\mu} \sin \mu \cos \lambda + \dot{\lambda} \cos \mu \sin \lambda))^T$$

La prima espressione è quella del nostro drone mentre la seconda quella relativa al payload.

Per quanto concerne il termine della rotazione va specificato da subito che la validità del modello è verificata solo per piccole variazioni di assetto del quadricottero secondo cui, quindi, è lecito pensare che le velocità angolari intorno a ciascun asse siano circa uguali alla derivate degli angoli di rotazione attorno agli assi stessi, ovvero:

$$\omega = (p, q, r)^T \sim (\dot{\phi}, \dot{\theta}, \dot{\psi})^T$$

Sarà perciò logico aspettarsi comparire nelle equazioni, non le componenti di ω (p, q, r), bensì gli angoli di Eulero (ϕ, θ, ψ).

I termini del tensore di inerzia diagonale "I" sono un dato del nostro problema e sono schematizzati con i valori "Jt" e "Ja". Non rimane che sostituire all'interno della formula iniziale ed ottenere:

$$K = \frac{1}{2} (m + m_B) (\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + \frac{1}{2} [J_t (\dot{\phi}^2 + \dot{\theta}^2) + J_a \dot{\psi}^2] + \frac{1}{2} m_B l^2 (\dot{\lambda}^2 \cos^2 \mu + \dot{\mu}^2) + m_B l [\dot{x} (\dot{\lambda} \cos \mu \cos \lambda - \dot{\mu} \sin \mu \sin \lambda) + \dot{y} \dot{\mu} \cos \mu + \dot{z} (\dot{\mu} \sin \mu \cos \lambda + \dot{\lambda} \cos \mu \sin \lambda)]$$

Più semplice e breve invece la trattazione sull'energia potenziale.

Quest'ultima infatti dipende proporzionalmente dalle masse in gioco e dalle loro rispettive altezze dal suolo. Se per il drone imponiamo un valore "z" di altitudine, guardando la terza componente relativa alla posizione del payload si ha facilmente:

$$z_B = z - l \cos \mu \cos \lambda$$

Si può infine esprimere l'energia potenziale del sistema come somma delle energie potenziali delle due masse:

$$U = - (m + m_B) g z - m_B g l \cos \lambda \cos \mu$$

Ora che sono caratterizzate le espressioni generali delle energie del nostro sistema possiamo introdurre la formulazione di Lagrange:

$$\frac{\partial}{\partial t} \left(\frac{\partial K}{\partial \dot{q}_i} \right) - \frac{\partial K}{\partial q_i} + \frac{\partial U}{\partial q_i} = Q_i$$

dove "K" e "U" rappresentano grandezze appena descritte.

La notazione "qi" sta ad indicare la variabile di stato secondo cui sono derivate le espressioni.

Saranno presenti tante equazioni di Lagrange tante quante sono le variabili di stato del nostro sistema, cioè tanti quanti sono i gradi di libertà.

Si otterranno quindi otto equazioni secondo le variabili $q_i = (x, y, z, \phi, \theta, \psi, \lambda, \mu)$.

Infine "Qi" rappresenta la componente della forza generalizzata del grado di libertà preso in considerazione.

Per quanto riguarda la trattazione di questo termine, v'è fatto un discorso a sè stante.

Viene infatti utilizzata la definizione di lavoro di un sistema.

Una possibile definizione è che la variazione di lavoro è proporzionale alla forza applicata in un punto per il vettore spostamento del punto di applicazione, in sintesi:

$$\Delta L = F \times \Delta s$$

Come è stato fatto per le energie cinetica e potenziale, anche per il lavoro vale un previo calcolo generale in una forma estesa e poi considerata nelle sue componenti all'interno delle singole equazioni. Per effettuare il giusto passaggio dal lavoro alla forza viene calcolato il lavoro in forma infinitesimale.

Le forze e i momenti in gioco sono tutti considerati dipendenti dalla motorizzazione, in particolare dal quadrato del regime di rotazione dei singoli motori similmente al calcolo dei momenti effettuati nella precedente simulazione.

Tutte le forze sono riportate dal sistema assi corpo alla terna fissa terrestre tramite il vettore:

$$k_B = (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi, \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi, \cos\phi \cos\theta)$$

Calcolando il lavoro infinitesimo si applica la semplificazione dovuta all'assunzione di piccole variazioni di angoli di rollio e beccheggio, ϕ e θ .

Per cui si ottiene:

$$\cos\phi \sim 1 \quad \sin\phi \sim \phi$$

$$\cos\theta \sim 1 \quad \sin\theta \sim \theta$$

$$k_B \sim (\theta \cos\psi + \phi \sin\psi, \theta \sin\psi - \phi \cos\psi, 1 - \phi^2 - \theta^2)^T + O(\theta^3, \phi^3)$$

I termini di second'ordine sulla terza componente non sono trascurati per tener conto così dell'effetto delle manovre di rollio e beccheggio, seppur piccole, sulla posizione della massa sospesa. Considerato questo aspetto possiamo scrivere finalmente l'equazione del lavoro:

$$\delta L = -(T_1 + T_2 + T_3 + T_4) [(\theta \sin\psi + \phi \cos\psi) \delta x + (\theta \sin\psi - \phi \cos\psi) \delta y + (1 - \phi^2 - \theta^2) \delta z] + (T_2 - T_4) \frac{d}{2} \delta \phi + (T_1 - T_3) \frac{d}{2} \delta \theta + (C_1 - C_2 + C_3 - C_4) \delta \psi$$

In particolare possiamo distinguere quattro termini principali all'interno dell'espressione del lavoro:

- termine della spinta totale dei quattro propulsori come già detto riportata in assi terrestri; (1)
- termine dovuto alla coppia generata dalla rotazione dei propulsori; (2)
- termine dovuto al momento generato dalla variazione della velocità di una coppia di rotori; (3)
- termine identico generato dalla seconda coppia di rotori; (4)

In questo modo possiamo ottenere ciascun termine di "Qi" relativa ad ogni equazione lagrangiana espressa in una variabile di stato. Le grandezze "T*" e "C*" rappresentano la dipendenza quadratica delle spinte e delle coppie dal regime di ciascun motore. Vengono ivi utilizzate come in precedenza costanti di spinta e di coppia denominati "Kt" e "Ks" rispettivamente.

A questo punto sono noti perfettamente tutti i fattori in gioco e possiamo passare alla risoluzione vera e propria del sistema. Per il calcolo si può suddividere il sistema in tre gruppi principali:

- calcolo secondo le variabili di posizione x,y,z;
- calcolo secondo le variabili di assetto ϕ, θ, ψ ;
- calcolo secondo le variabili del payload λ, μ ;

Ciascun valore incognito come sarà poi visibile sarà espresso in forma di derivata seconda.

Essenzialmente ciascun gruppo di variabili possiede una forma matematica simile dell'equazione di Lagrange. A questo punto si introducono le equazioni di Lagrange per lo spostamento x, y, z:

$$\begin{aligned}\ddot{x} + \sigma l (\ddot{\lambda} \cos\mu \cos\lambda - \ddot{\mu} \sin\mu \sin\lambda) &= \sigma l [2\dot{\lambda}\dot{\mu} \sin\mu \cos\lambda + (\dot{\lambda}^2 + \dot{\mu}^2) \cos\mu \sin\lambda] - \tau g (\theta \cos\psi - \phi \sin\psi) \\ \ddot{y} + \sigma l \ddot{\mu} \cos\mu &= \sigma l \dot{\mu}^2 \sin\mu - \tau g (\theta \sin\psi - \phi \cos\psi) \\ \ddot{z} + \sigma l (\ddot{\mu} \sin\mu \cos\lambda + \ddot{\lambda} \cos\mu \sin\lambda) &= \sigma l [2\dot{\lambda}\dot{\mu} \sin\mu \sin\lambda - (\dot{\lambda}^2 + \dot{\mu}^2) \cos\mu \cos\lambda] + g [1 - \tau (1 - \phi^2 - \theta^2)]\end{aligned}$$

Dove i valori di " σ " e " τ " sono grandezze introdotte al fine di compattare la forma delle equazioni e di rendere meglio visibile il legame tra le varie incognite del sistema. In particolare:

$$\sigma = \frac{m_b}{(m + m_B)}$$

" σ " rappresenta un parametro adimensionale che esprime la percentuale in peso del payload rispetto alla massa del sistema totale.

$$\tau = \frac{T}{(mg)}$$

dove

$$T = T_1 + T_2 + T_3 + T_4 = K_t (\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)$$

" τ " invece è il rapporto spinta-peso del nostro quadricottero senza il carico aggiuntivo.

Ovviamente viene riutilizzato il legame quadratico tra spinta e velocità di rotazione dei rotori.

Questo per ottenere maggior chiarezza e compattezza nella forma e permettere di meglio visualizzare il legame tra le incognite.

Il successivo set di equazioni considerato è quello riguardante gli angoli di Eulero che esprimiamo:

$$J_t \ddot{\phi} = (T_2 - T_4) \left(\frac{d}{2} \right)$$

$$J_t \ddot{\theta} = (T_1 - T_3) \left(\frac{d}{2} \right)$$

$$J_a \ddot{\psi} = (C_1 - C_2 + C_3 - C_4)$$

Se si osserva l'espressione dell'energia cinetica del sistema si nota che la dipendenza degli angoli di Eulero è espressa soltanto nel termine di energia cinetica rotazionale sotto forma di velocità angolare, ovvero sotto forma di derivata prima. Ne consegue che le tre equazioni appena descritte sono singolarmente disaccoppiabili dal resto del problema in quanto sono risolvibili in maniera autonoma. Non restano che le rimanenti due equazioni delle variabili del carico sospeso:

$$\ddot{\lambda} \cos^2 \mu + (\ddot{x} \cos \mu \cos \lambda + \ddot{z} \cos \mu \sin \lambda) \frac{1}{l} = 2\dot{\lambda}\dot{\mu} \sin \mu \cos \mu - \omega^2 \sin \lambda \cos \mu$$

$$\ddot{\mu} - (\ddot{x} \sin \mu \sin \lambda - \ddot{y} \cos \mu - \ddot{z} \sin \mu \cos \lambda) \frac{1}{l} = -\omega^2 \cos \lambda \sin \mu$$

Dove " ω " :

$$\omega = \sqrt{\frac{g}{l}}$$

rappresenta frequenza naturale di un pendolo semplice di lunghezza data " l ".

Questa modellazione appare consistente nel caso in cui si assumono trascurabili gli effetti della resistenza sul moto oscillatorio del carico. Quindi i carichi inerziali dovuti semplicemente alla proprietà di massa del pendolo si considerano molto superiori agli effetti della resistenza dell'aria durante il moto.

3.3 Modello Linearizzato:

Una volta conosciuta la fisica dietro il nostro problema ci si è resi conto che non è possibile utilizzare le equazioni introdotte in maniera semplice, ovvero ottenere prima una forma esplicita e in seguito immettere i dati per ottenere i risultati.

Infatti, a parte le equazioni dell'assetto, tutte le altre presentano una forte dipendenza non solo di tutte le incognite presenti ma anche delle loro derivate prime e seconde il che rende necessario un ulteriore gradino prima di arrivare all'implementazione al calcolatore.

Scoppiando il sistema dalle equazioni ϕ , θ , ψ si è proceduto ad un'integrazione numerica del sistema restante costituito dalle restanti cinque al fine di ridurre la complessità indicata soprattutto dalla dipendenza dalle derivate seconde espresse nella parte a sinistra delle espressioni (che riportiamo nuovamente per chiarezza):

$$\ddot{x} + \sigma l (\ddot{\lambda} \cos \mu \cos \lambda - \ddot{\mu} \sin \mu \sin \lambda) = \sigma l [2\dot{\lambda}\dot{\mu} \sin \mu \cos \lambda + (\dot{\lambda}^2 + \dot{\mu}^2) \cos \mu \sin \lambda] - \tau g (\theta \cos \psi - \phi \sin \psi)$$

$$\ddot{y} + \sigma l \ddot{\mu} \cos \mu = \sigma l \dot{\mu}^2 \sin \mu - \tau g (\theta \sin \psi - \phi \cos \psi)$$

$$\ddot{z} + \sigma l (\ddot{\mu} \sin \mu \cos \lambda + \ddot{\lambda} \cos \mu \sin \lambda) = \sigma l [2\dot{\lambda}\dot{\mu} \sin \mu \sin \lambda - (\dot{\lambda}^2 + \dot{\mu}^2) \cos \mu \cos \lambda] + g [1 - \tau (1 - \phi^2 - \theta^2)]$$

$$\ddot{\lambda} \cos^2 \mu + (\ddot{x} \cos \mu \cos \lambda + \ddot{z} \cos \mu \sin \lambda) \frac{1}{l} = 2\dot{\lambda}\dot{\mu} \sin \mu \cos \mu - \omega^2 \sin \lambda \cos \mu$$

$$\ddot{\mu} - (\ddot{x} \sin \mu \sin \lambda - \ddot{y} \cos \mu - \ddot{z} \sin \mu \cos \lambda) \frac{1}{l} = -\omega^2 \cos \lambda \sin \mu$$

Il vantaggio consiste nella possibilità di esprimere suddette parti di sinistra in seguito con una matrice 5X5 che mostra un costo computazionale visibilmente minore:

$$\begin{bmatrix} 1 & 0 & 0 & \sigma l \cos \mu \cos \lambda & -\sigma l \sin \mu \sin \lambda \\ 0 & 1 & 0 & 0 & \sigma l \cos \mu \\ 0 & 0 & 1 & \sigma l \cos \mu \sin \lambda & \sigma l \sin \mu \cos \lambda \\ (\cos \mu \cos \lambda)/l & 0 & (\cos \mu \sin \lambda)/l & \cos^2 \mu & 0 \\ -(\sin \mu \sin \lambda)/l & (\cos \mu)/l & (\sin \mu \cos \lambda)/l & 0 & 1 \end{bmatrix}$$

Sarà definita la matrice come "K". I rimanenti elementi delle espressioni, quelli di destra, sono invece riuniti in una matrice colonna 5X1 che d'ora in avanti chiameremo "G".

La si indica perciò come:

$$\begin{aligned}
& \left[\begin{aligned}
& \sigma l [2\dot{\lambda}\dot{\mu} \sin\mu \cos\lambda + (\dot{\lambda}^2 + \dot{\mu}^2) \cos\mu \sin\lambda] - \tau g (\theta \cos\psi + \phi \sin\psi) \\
& \sigma l \dot{\mu}^2 \sin\mu - \tau g (\theta \sin\psi - \phi \cos\psi) \\
& \sigma l [2\dot{\lambda}\dot{\mu} \sin\mu \sin\lambda + (\dot{\lambda}^2 + \dot{\mu}^2) \cos\mu \cos\lambda] - \tau g (\theta \sin\psi - \phi \cos\psi) \\
& 2\dot{\lambda}\dot{\mu} \sin\mu \cos\mu - \omega^2 \sin\lambda \cos\mu \\
& -\omega^2 \cos\lambda \sin\mu
\end{aligned} \right]
\end{aligned}$$

Con questa notevole riduzione è possibile scrivere una nuova forma più compatta per quanto riguarda il set di cinque equazioni di Lagrange:

$$[K][\ddot{x}, \ddot{y}, \ddot{z}, \ddot{\lambda}, \ddot{\mu}]^T = [G]$$

ovvero, invertendo la matrice K:

$$[\ddot{x}, \ddot{y}, \ddot{z}, \ddot{\lambda}, \ddot{\mu}]^T = [K]^{-1} [G]$$

Il sistema risolutivo finale per la risoluzione del moto del quadricottero e annesso pendolo risulta:

$$\left\{ \begin{aligned}
& [\ddot{x}, \ddot{y}, \ddot{z}, \ddot{\lambda}, \ddot{\mu}]^T = [K]^{-1} [G] \\
& J_t \ddot{\phi} = (T_2 - T_4) \left(\frac{d}{2} \right) \\
& J_t \ddot{\theta} = (T_1 - T_3) \left(\frac{d}{2} \right) \\
& J_a \ddot{\psi} = (C_1 - C_2 + C_3 - C_4)
\end{aligned} \right.$$

Sarà questo sistema che si utilizzerà per la successiva implementazione.

Cap.4: Implementazione in Matlab e Simulink

4.1 Implementazione Generale del Modello

L'implementazione seguente è stata svolta utilizzando sempre la piattaforma Matlab/Simulink.

Questa volta però non ci si è preoccupati di un confronto tra due versioni implementate in modi diversi ma si è ottenuto un modello che utilizza entrambi gli strumenti del software al fine di descrivere la dinamica di un sistema massa-pendolo.

Il primo passo consiste sempre nell'introduzione dei parametri fissi o comunque conosciuti.

E' stato quindi implementato uno script, denominato "SCRIPTLANCIO", dove vengono inclusi i parametri fisici e geometrici del nostro sistema massa-pendolo:

- massa del velivolo " M ";
- massa del payload " M_b ";
- distanza interasse del drone " d ";
- lunghezza cavo del pendolo " l ";
- elementi della matrice di inerzia " J_t ", " J_a ";
- costante di spinta " K_t ";
- costante di coppia " K_s ";
- rapporto tra peso del drone e peso totale del sistema " σ ";
- frequenza di oscillazione del pendolo " ω ";

Successivamente si è implementato uno schema a blocchi in Simulink dove, per cominciare, venivano introdotte e trovate le ulteriori grandezze necessarie.

Questo passaggio è stato realizzato in un sottosistema chiamato "VARIABILI TEMPORALI".

Benché non sia stata considerata l'implementazione della resistenza in chiave dinamica, viste le piccole manovre a cui il drone è sottoposto, si è pensato di creare una modellazione completa di una manovra dove la spinta non rappresenta un parametro fisso come nelle simulazioni precedenti.

I regime di rotazione dei rotori quindi dovranno avere dei precisi valori in precisi intervalli temporali per permettere la giusta evoluzione fino allo stato iniziale.

Quindi la soluzione migliore è stata adottare dei segnali a gradino in cui sono stati introdotti manualmente i punti costruendo progressivamente la manovra desiderata. Sostanzialmente questi segnali dovranno "guidare" il quadricottero nella maniera corretta.

Con $\omega_1, \omega_2, \omega_3, \omega_4$ funzioni del tempo anche tutte le variabili da loro dipendenti risulteranno anch'esse funzioni del tempo:

-il rapporto spinta/peso τ ;

-le spinte T_1, T_2, T_3, T_4 ;

-le coppie C_1, C_2, C_3, C_4 ;

I valori ivi elencati rappresentano le uscite del nostro primo sottosistema.

Ora si conosce tutto l'occorrente per poter implementare il nostro set completo di otto equazioni.

A questo punto le ultime uscite e quelle lanciate per mezzo dello script iniziale sono messe a disposizione tramite chiamata dal Workspace con il blocco "constant": in particolare confluiscono nel sottosistema principale "SISTEMA 8GDL" per il calcolo.

Come già annunciato nel modello implementato il sistema viene scoppiato in due parti e per il singolo svolgimento di ciascuna è stata scritta una "Matlab Function" in SISTEMA 8GDL.

Questo blocco permette di editare di una funzione sotto forma di m.file, come uno script, e di crearne un blocco dove l'utente inserisce input ed output in Simulink.

Nella prima function si risolvono le equazioni di Lagrange relative a x, y, z, λ, μ tramite le matrici K e G definite nello script stesso come nel capitolo precedente.

Essendo inoltre un prodotto matriciale l'uscita dal blocco è un singolo vettore colonna, in particolare viste le dimensioni di K e G , di cinque elementi. Come è noto, questi valori rappresentano le derivate seconde delle relative equazioni di riferimento e necessitano di successive integrazioni. Quest'ultimo step è effettuato con il blocchetto "integrator" cui devono essere apportate le giuste condizioni iniziali del problema. Sono presenti perciò due integratori a cascata per ottenere il vettore contenente le cinque incognite cercate: x, y, z, λ e μ .

La seconda function, più semplice, prevede il calcolo allo stesso modo degli angoli di ϕ, θ e ψ del drone utilizzando le tre lagrangiane relative prima introdotte.

L'out del secondo blocco quindi sarà sempre in forma vettoriale, questa volta logicamente rappresentata da un vettore colonna di tre elementi. Le due successive integrazioni renderanno infine disponibili gli angoli di assetto allo stesso modo. I valori in uscita inoltre non sono finì a sè stessi ma sono di nuovo utilizzati in un loop chiuso. Le condizioni iniziali negli integratori permettono il calcolo delle otto grandezze nel primo intervallo temporale e successivamente ciascun risultato viene riutilizzato per l'integrazione nell'intervallo successivo fino alla fine della simulazione consentendo un'evoluzione in tempo reale di ciascuna incognita.

Il loop agisce nelle definizioni della matrici K e G dove si trovano tra gli ingressi tre parametri non precedentemente introdotti:

- "VAR" che rappresenta il vettore contenente x, y, z, λ e μ ;
- "VARDER" che invece contiene le derivate prime di x, y, z, λ e μ ;
- "ANG" che infine contiene i valori di φ, θ e ψ .

Al termine della simulazione con dei classici blocchetti "scope" sarà possibile vedere i risultati.

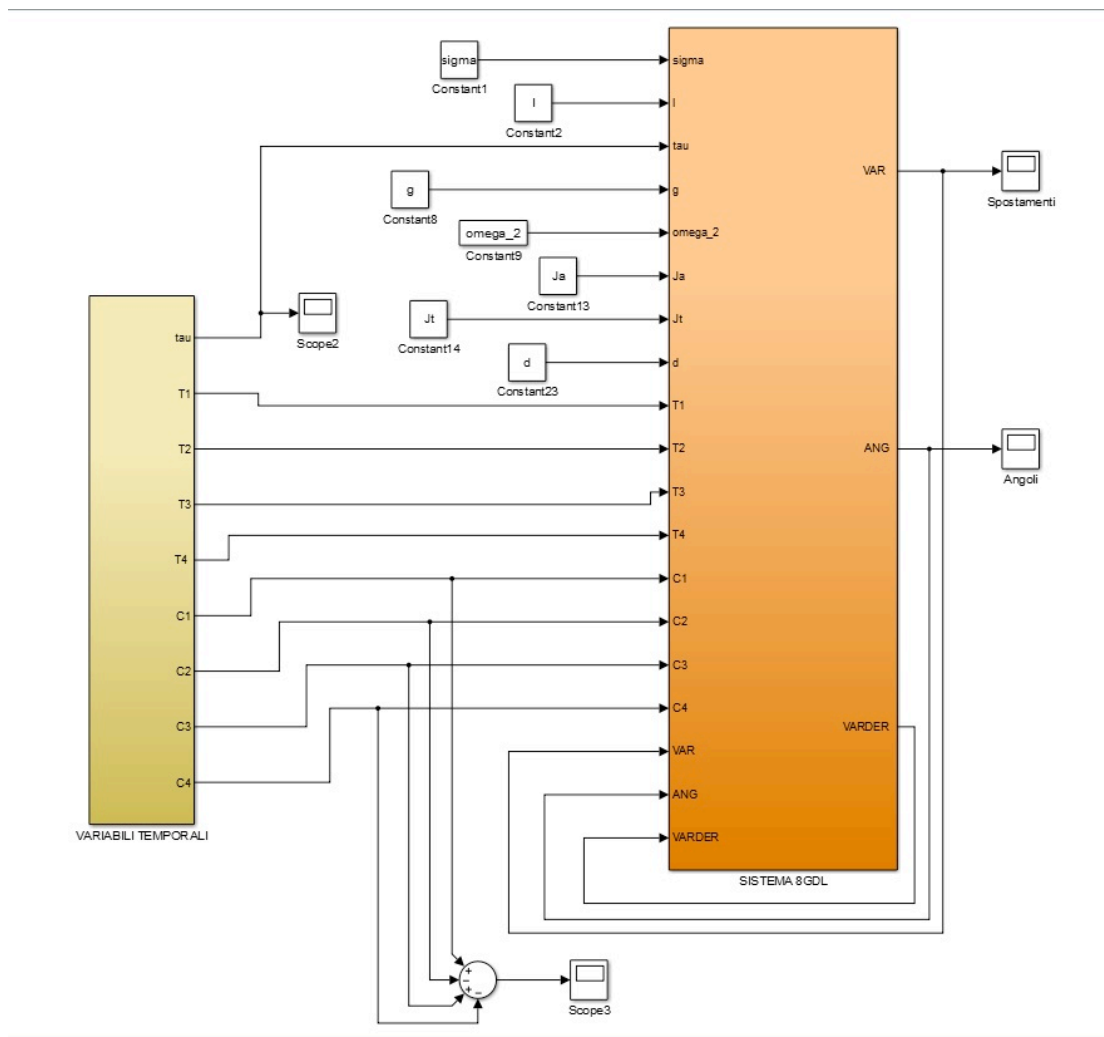


Fig.33:Il modello implementato in Simulink

4.2 Manovre simulate

Da quello che si è potuto intravedere in questo lavoro, è chiaro oramai che i parametri d'immediato utilizzo da parte dell'utente sono la gestione delle manette delle motorizzazioni istallate.

Per avere risultati di maggiore complessità e realismo sono state questa volta utilizzate funzioni del tempo per descrivere l'andamento dei regimi di rotazione caso per caso.

Le funzioni sono state scritte per punti, rappresentando quindi dei segnali a gradino o rampe.

La simulazione in questo modo riguarda non il mantenimento di uno stato preimpostato, che sia di avanzamento o di hovering, del drone ma la rotazione attorno ad un asse a partire da una condizione iniziale stabile fino al ritorno a quest'ultima o comunque ad un'altra condizione stabile seguendo la sua evoluzione.

La condizione iniziale per semplicità è quella di hovering.

A partire da questa si è provato ad effettuare perciò una movimentazione dei singoli angoli di assetto in tre distinte parti gestendo ogni volta i motori seguendo le regole di pilotaggio necessarie descritte nella descrizione fisica dell'assetto e della meccanica del volo.

Seguiranno movimenti completi descriventi manovre di rollio, beccheggio e imbardata rispettando i limiti imposti dal modello di piccole variazioni.

Le simulazioni avranno una durata di 50s o 100s a seconda della variazione della grandezze misurate: il tempo deve essere sufficiente a mostrare infatti una nuova stabilità raggiunta.

Si ricordi inoltre che il legame tra spinta e rotazione dei motori è quadratico quindi egual variazioni di manetta, ma opposte, su ciascuna coppia o motore non producono un effetto simmetrico o comunque a senso unico sul velivolo.

Per questo la manovra, se non controllata, mostra sempre un effetto secondario da limitare.

Infatti se guardiamo il parametro τ , si nota che esso malgrado tutto non sempre è assolutamente costante nelle simulazioni.

Senza l'utilizzo di sistemi di controllo sarà molto spesso obbligatorio trovare una soluzione di compromesso.

4.2.1 Manovra in Hover

La prima manovra cercata è stata quella di hovering, principalmente atta a testare la validità del lavoro implementato. Nonostante la presenza del pendolo non si ha una variazione nella dinamica dello svolgimento del comando rispetto al caso precedente. Il payload, infatti, è fermo e lo continuerà ad essere per tutta la durata della simulazione per definizione della manovra stessa.

L'unica differenza rispetto allo studio precedente, oltre alle equazioni utilizzate, è data dalla manetta che deve logicamente essere maggiorata per sopperire al peso del quadricottero e del carico ad esso appeso. Tant'è vero che le condizioni iniziali rispecchiano la simulazione eseguita con le equazioni di Eulero-Newton:

$$\begin{bmatrix} x, y, z, \lambda, \mu \end{bmatrix}^T = 0 \qquad \begin{bmatrix} \phi, \theta, \psi \end{bmatrix}^T = 0$$

Significa velocità in assi terrestri nulla, assetto nullo con il drone messo in piano e pendolo fermo e tali dovranno rimanere queste grandezze nel corso di tutta la durata della prova.

L'assetto ottenuto, intuitivamente, è costante:

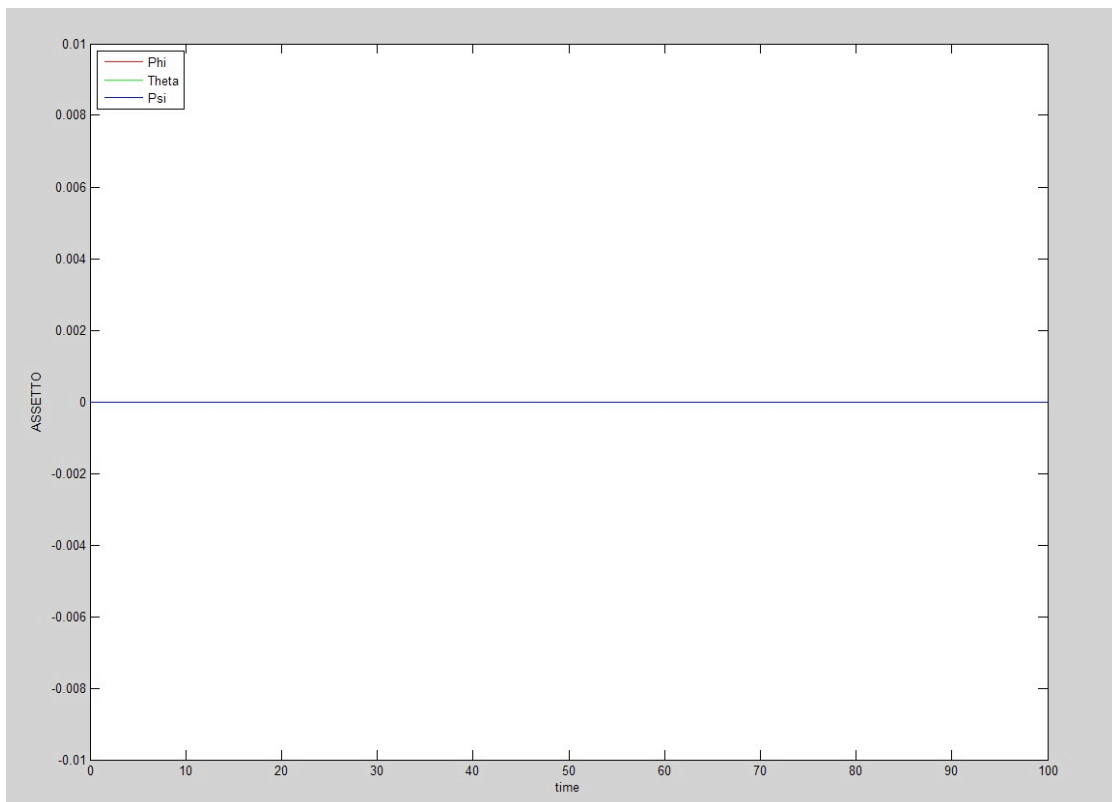


Fig.34: Assetto stabile in Hovering.

Facilmente il segnale rappresentante i motori è pari ad un valore costante anch'esso:

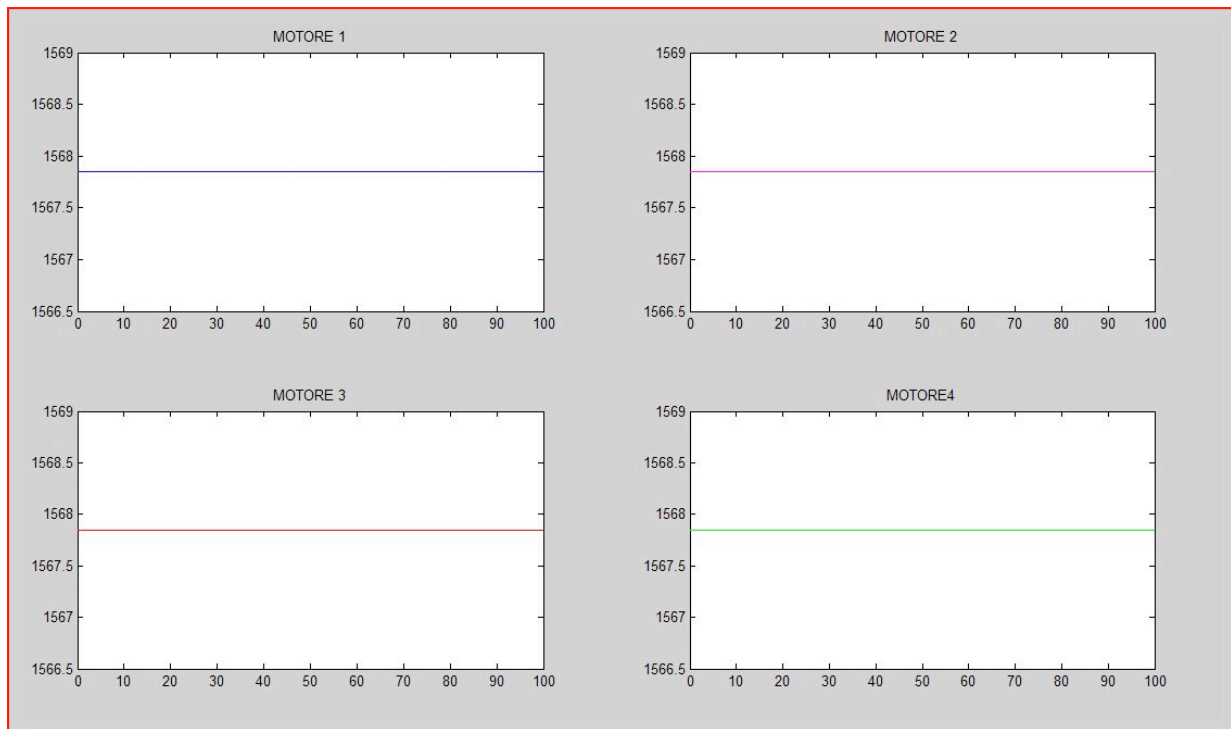


Fig.35: I motori girano tutti alla stessa velocità.

Il pendolo che in genesi era immobile resta tale come riscontrato in figura:

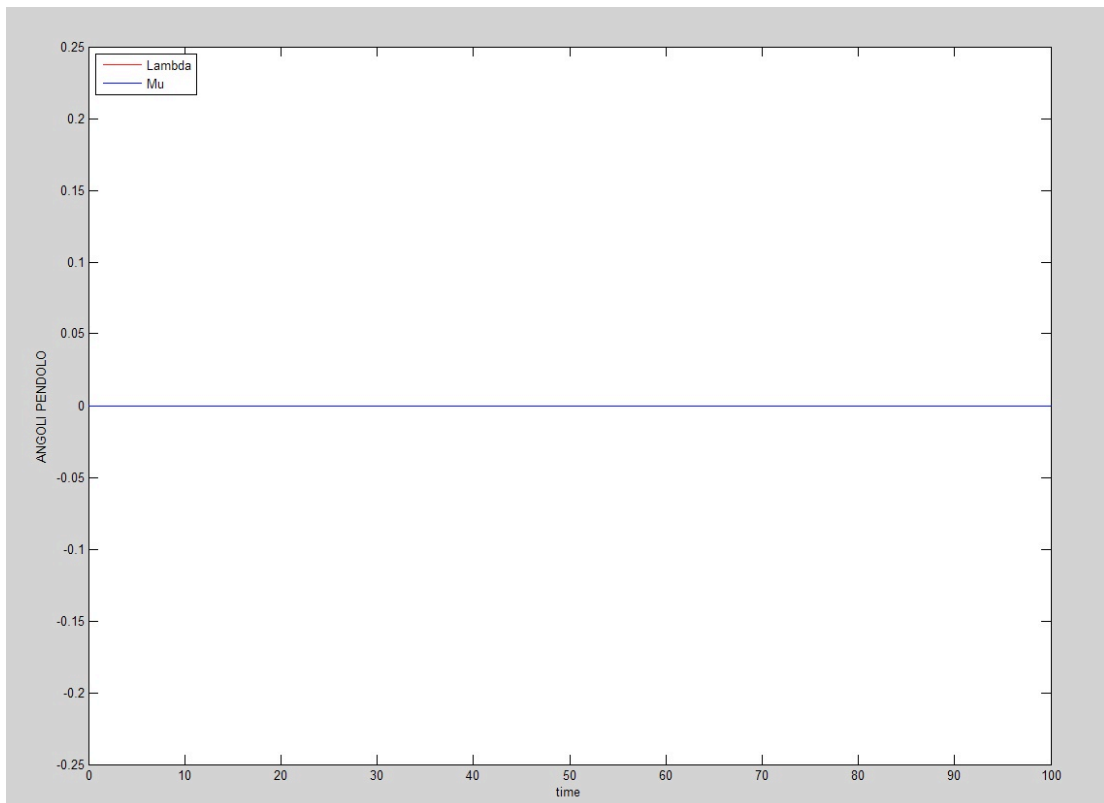


Fig.36: Plot di λ e μ .

Per definizione di volo a punto fisso, di nuovo, gli spostamenti anch'essi risultano nulli:

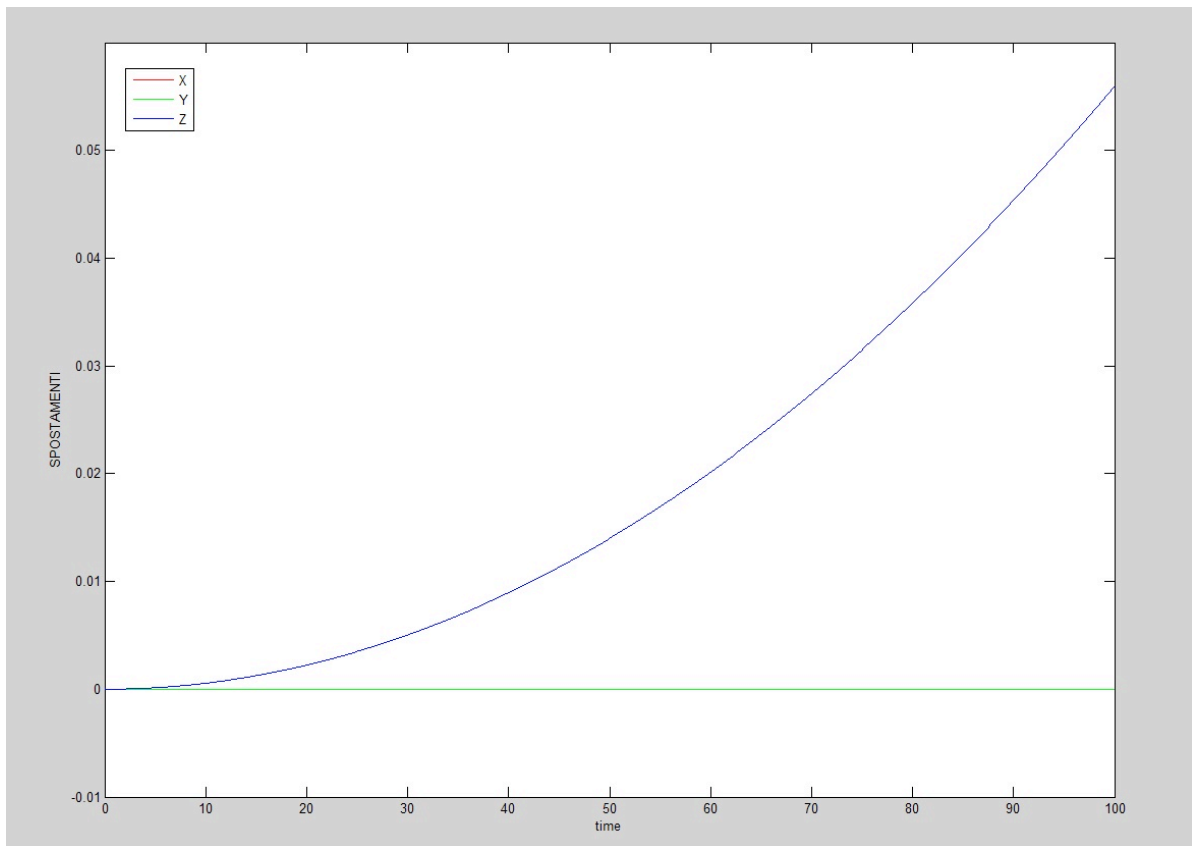


Fig.37: Gli spostamenti infinitesimi.

In questa condizione stazionaria di volo non ci sono effetti secondari dinamici e non sono di fatto stati riscontrati.

4.2.2 Manovra in Hovering con Pendolo in Movimento

Più interessante appare invece la situazione nel momento in cui introduciamo all'istante iniziale un valore diverso da zero ad uno dei due gradi di libertà del pendolo.

Infatti la presenza di una oscillazione fa sì che il quadricottero in posizione di hovering subisca gli effetti inerziali del movimento del carico. A seconda della direzione del movimento di quest'ultimo sono state rilevate diverse fluttuazioni periodiche anche nel posizionamento del drone.

La mancanza di resistenza non crea effetti di smorzamento sul pendolo che una volta avviato non è in grado di fermarsi. Nelle simulazioni sono stati introdotti prima un valore di λ pari a 10° (in rad)

$$[x, y, z, \lambda, \mu]^T = [0, 0, 0, 10 \cdot \frac{\pi}{180}, 0]^T$$

ed in seguito un valore analogo per μ .

$$[x, y, z, \lambda, \mu]^T = [0, 0, 0, 0, 10 \cdot \frac{\pi}{180}]^T$$

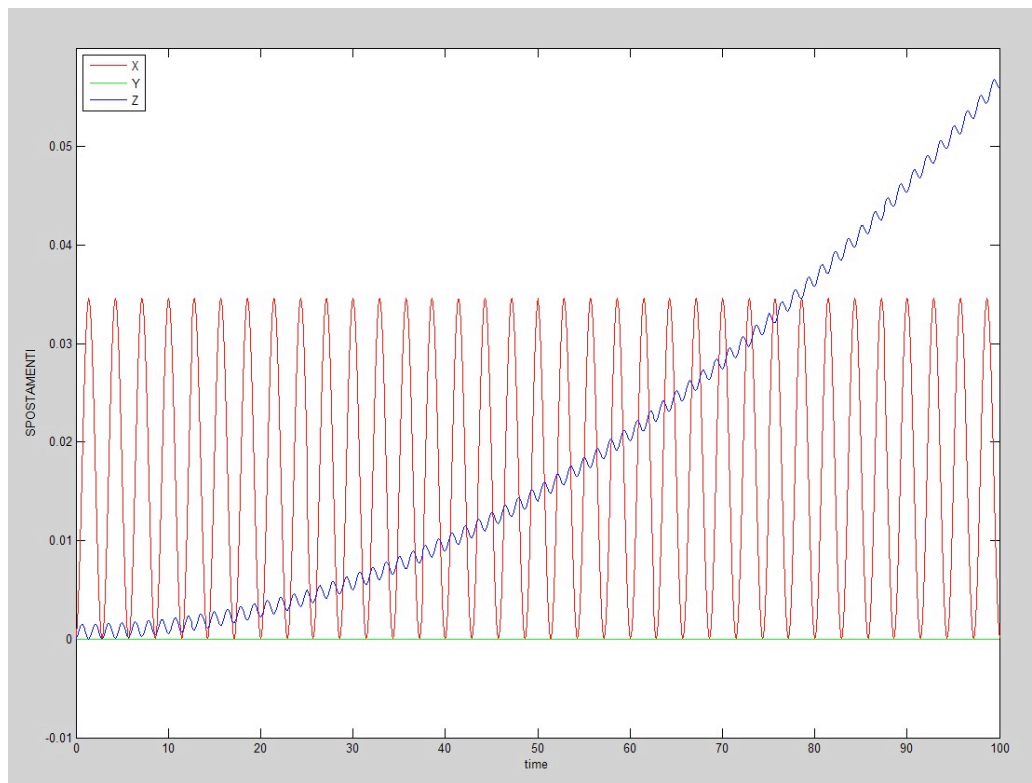


Fig.38: Gli spostamenti infinitesimi in presenza di oscillazione frontale.

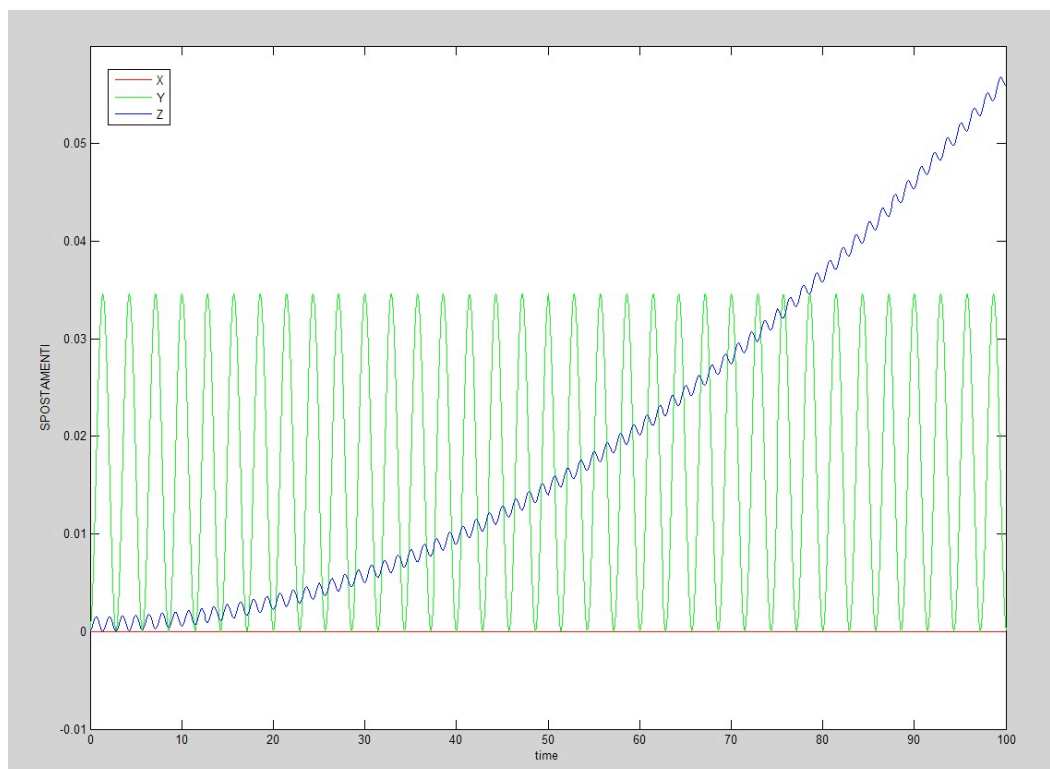


Fig.39: Gli spostamenti infinitesimi in presenza di oscillazione laterale.

Globalmente la fluttuazione del pendolo ha un effetto nullo sul valore di spostamento finale riscontrato rispetto al caso di pendolo immobile. L'assetto, come nel caso precedente, è rimasto stazionario quindi il pendolo non lo modifica in nessun modo:

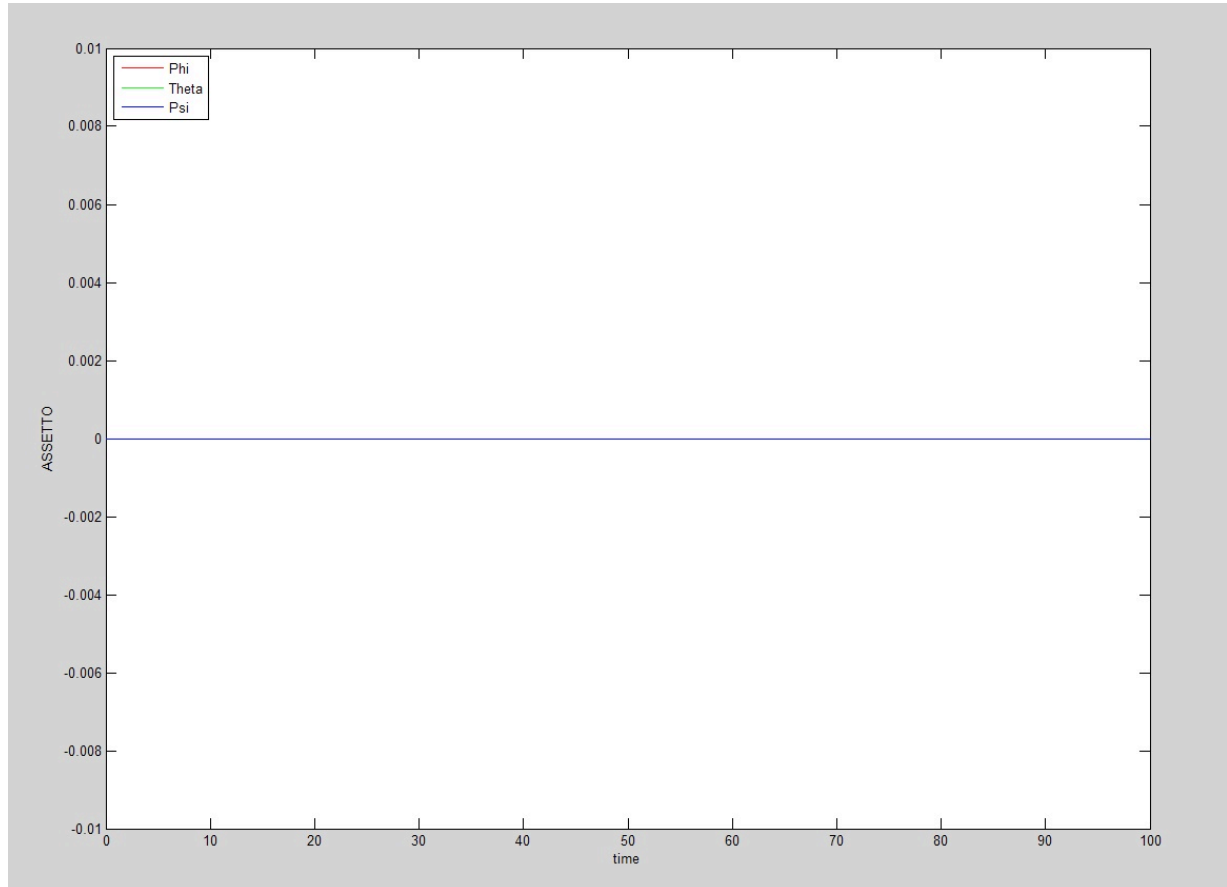


Fig.40: L'assetto, come mostrato, non cambia.

Si deduce perciò che il pendolo non modifica in maniera quantitativa nè assetto nè il valore numerico degli spostamenti. Le due equazioni del set che descrivono il moto del pendolo secondo i due gradi λ e μ invece hanno riportato semplicemente il moto di quest'ultimo continuo nel tempo.

Dopo aver appurato l'accuratezza del lavoro eseguito si è passati alla fase successiva simulando come già detto manovre intorno a φ , θ e ψ che saranno presentate nello stesso ordine.

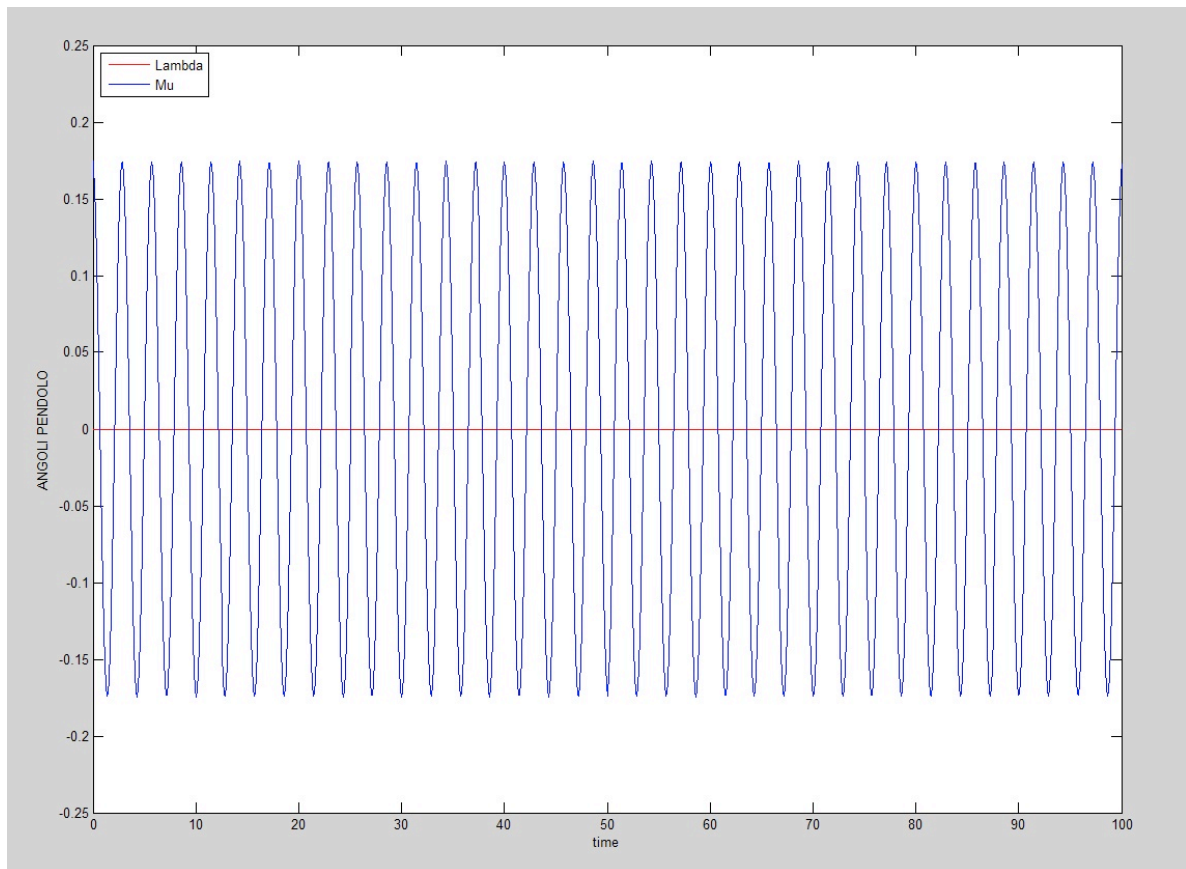
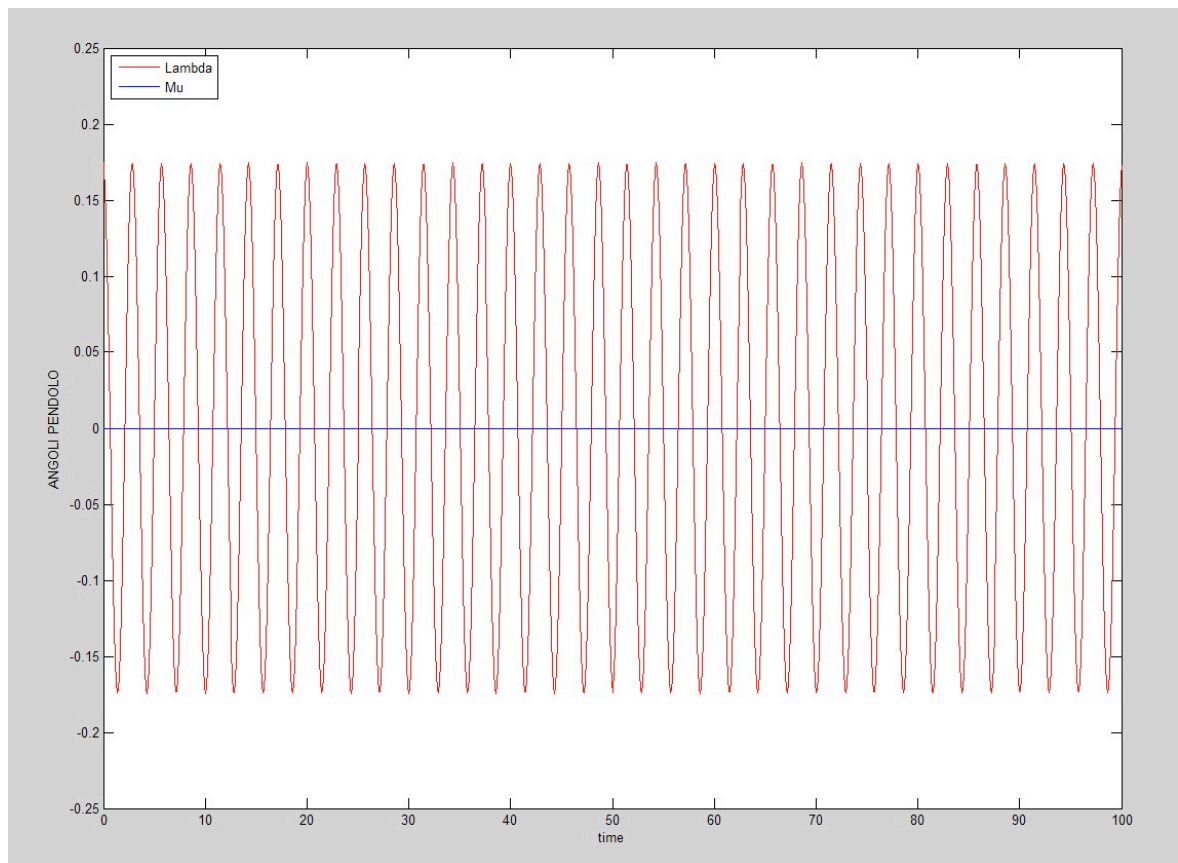


Fig.41-42:Rappresentazione del moto del pendolo.

4.2.3 Manovra in Rollio

La manovra di rollio è stata realizzata considerando la rotazione in entrambi i sensi.

Inoltre questa prevede una permanenza nello stato di inclinazione entrambe le volte.

Partendo ancora una volta da una condizione di hovering con pendolo fermo ed una altitudine preimpostata di 40m, il quadrirotore effettua una rollata negativa, verso destra, e dopo un breve stazionamento in assetto nullo effettua la simmetrica manovra verso sinistra:

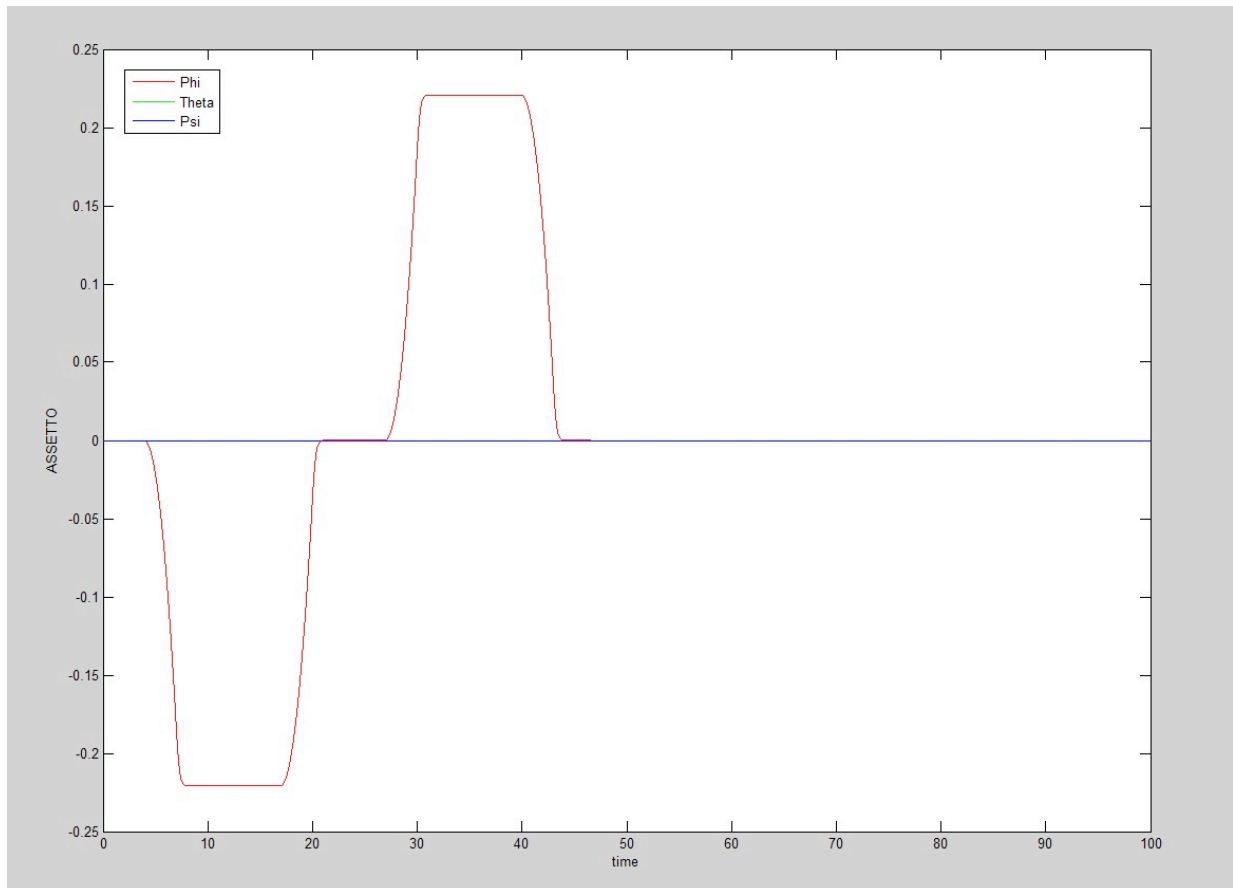


Fig.43: La manovra compiuta in rollio.

Ovviamente i segnali a gradino che rappresentano l'andamento dei motori sono stati pensati in modo opportuno da mantenere la quota anche quando il velivolo è inclinato maggiorando la spinta complessiva come appare visibile:

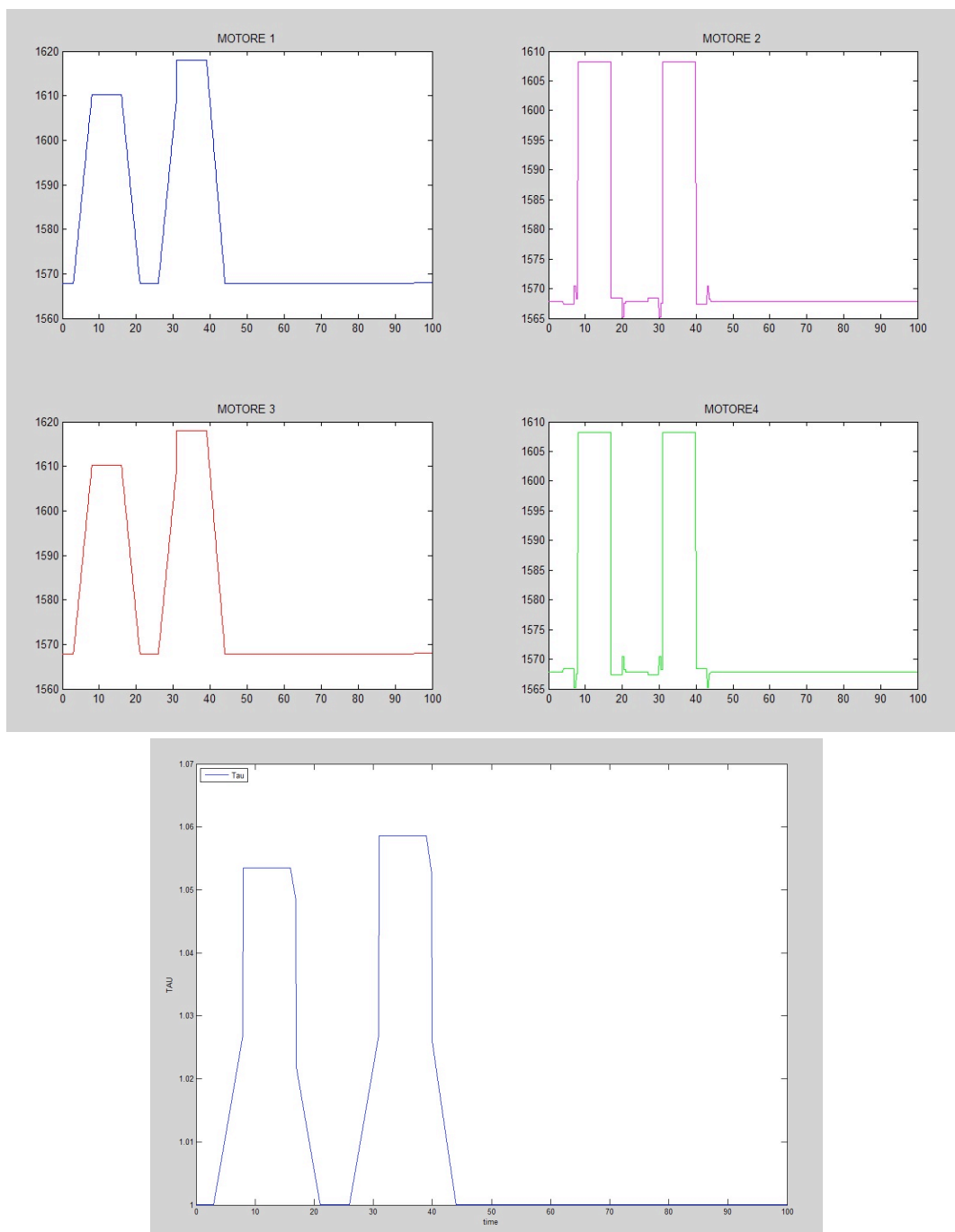


Fig.44-45: Rispettivamente, andamento dei motori e rapporto spinta/peso.

E' riscontrato che la quota, z , è stata complessivamente mantenuta a meno di piccole deviazioni da considerarsi nell'introduzione di segnali per punti.

Lo spostamento laterale, y , è da considerare in maniera particolare all'interno del modello a causa dell'assenza di resistenza. Infatti nel momento in cui comincia la virata si subisce l'accelerazione della componente non verticale della spinta ed, inoltre, finchè non inizia la contro-virata, anche nel momento in cui ho assetto nullo, non ho nessun freno aerodinamico. La velocità laterale permette ancora lo spostamento con lo stesso valore. Solo dopo, con una componente contraria dovuta

all'opposto angolo di rollio si può arrestare il drone e mantenere un valore costante fino alla fine della simulazione. Per chiarezza il grafico degli spostamenti:

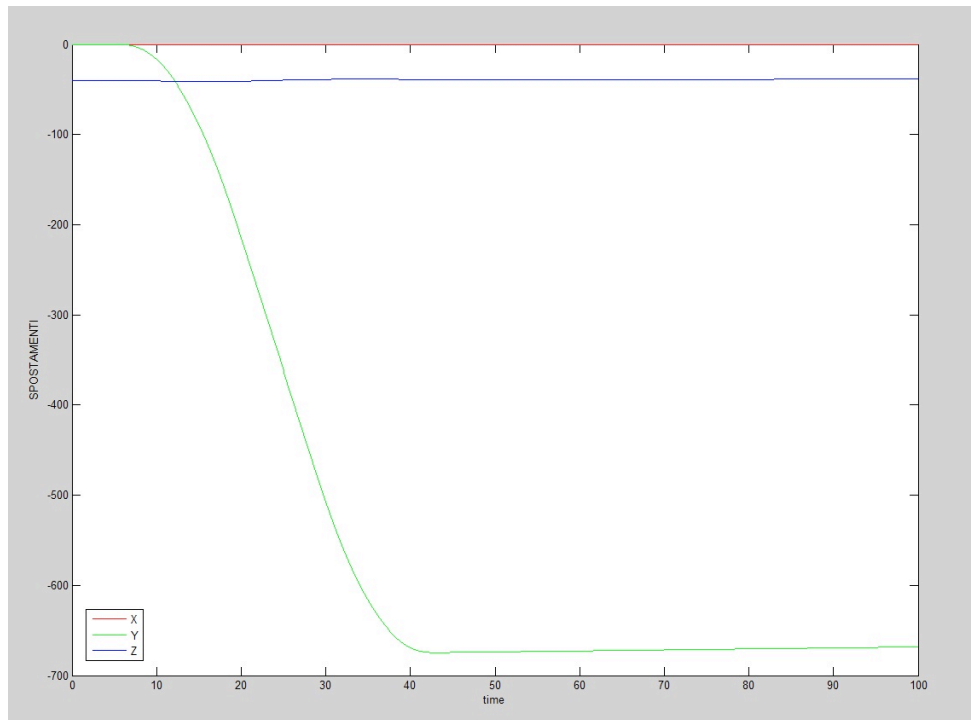


Fig.46: Notare soprattutto l'andamento di y che alla fine del secondo rollio è costante.

La complessità del pendolo è stata anch'essa descritta in maniera corretta dal modello:

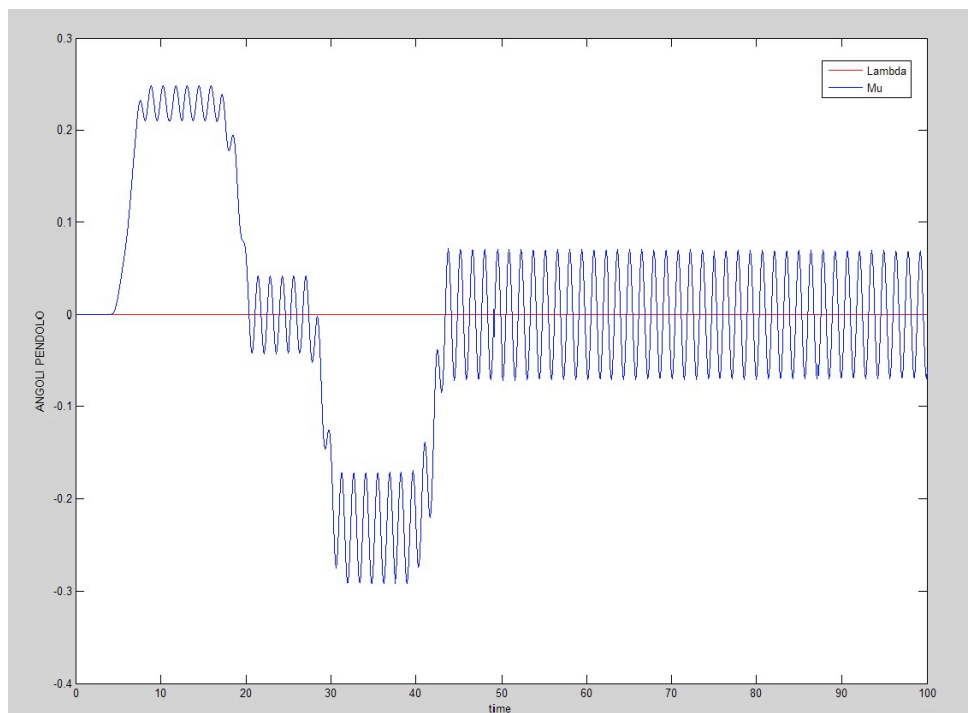


Fig.47: Il pendolo nel corso della simulazione

Da notare un movimento oscillatorio laterale dovuto alle due fasi distinte di manovra e l'oscillazione restante una volta che il veicolo si è messo nuovamente in posizione piana.

4.2.4 Manovra in Beccheggio

Il movimento in beccheggio è stato eseguito in maniera diversa dalla manovra precedente.

A questo proposito il drone è stato settato in maniera tale che si trovasse già in uno stato di assetto picchiato con una velocità iniziale “x” ad una altitudine “z “ ed in seguito si sia provveduto a riportarlo in piano. Conseguenza di questa scelta è che nella simulazione il velivolo accelera finchè mantiene lo stato iniziale e successivamente prosegue nella direzione frontale di moto rettilineo uniforme. Il pendolo viene dato con l'angolo di direzione frontale λ pari all'angolo di beccheggio θ poichè il pendolo rispetto alla direzione del versore verticale del velivolo ha sempre questo stesso angolo di inclinazione. Si riportano le condizioni iniziali:

$$[x, y, z, \lambda, \mu]^T = [2, 0, -20, -7 \cdot \frac{\pi}{180}, 0]^T$$

$$[\phi, \theta, \psi]^T = [0; -7 \cdot \frac{\pi}{180}; 0]^T$$

Viene mostrato prima di tutto tramite gli angoli di Eulero l'evoluzione dell'assetto:

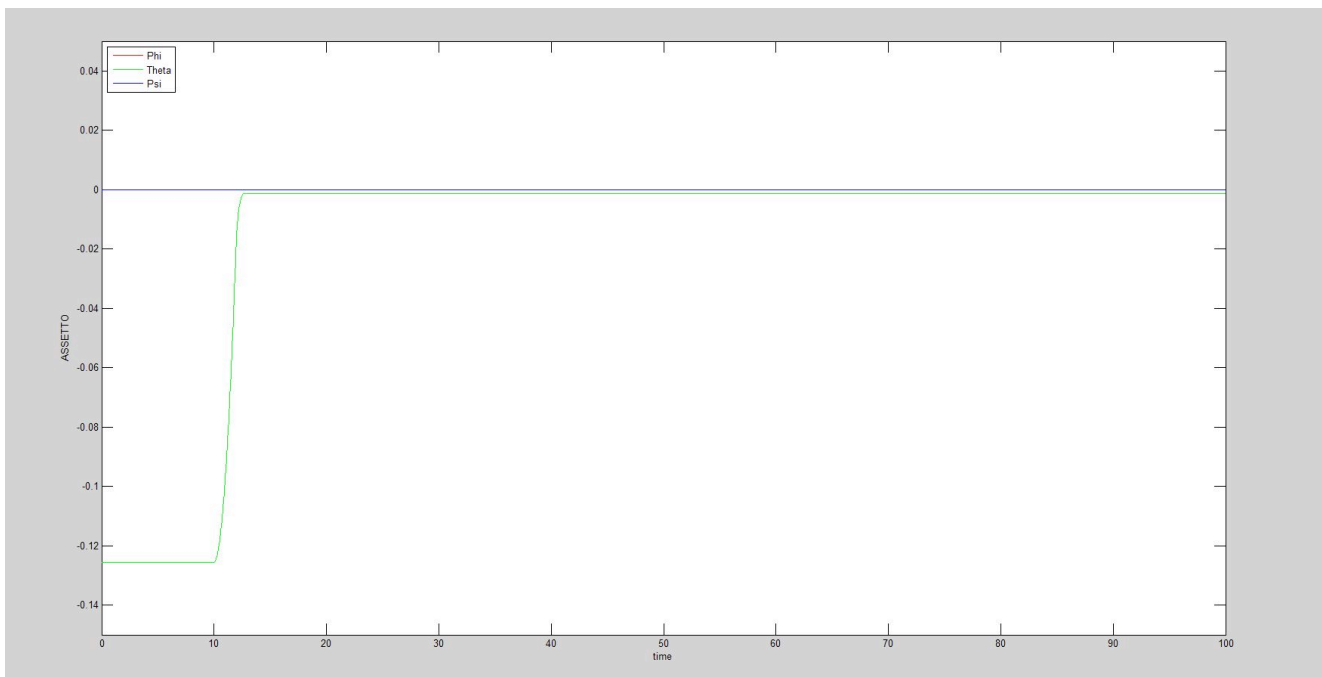


Fig.48: La cabrata rappresentata.

Come visibile il pilotaggio richiesto da una variazione di angolo di beccheggio coinvolge i motori opposti a quelli del rollio:

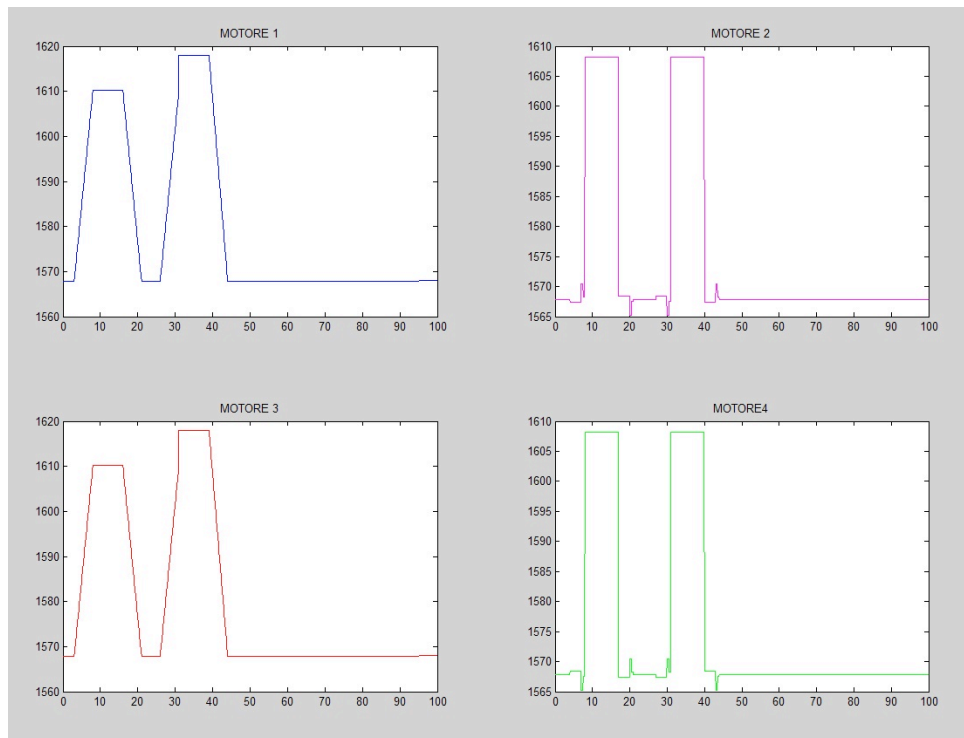


Fig.49: In questo caso sono coinvolti i motori 1-3.

Come ci si aspettava si assiste ad un andamento parabolico dello spostamento che rappresenta l'accelerazione. L'andamento rettilineo seguente indica che il velivolo continua a traslare causa la mancanza di resistenza una volta raddrizzato.

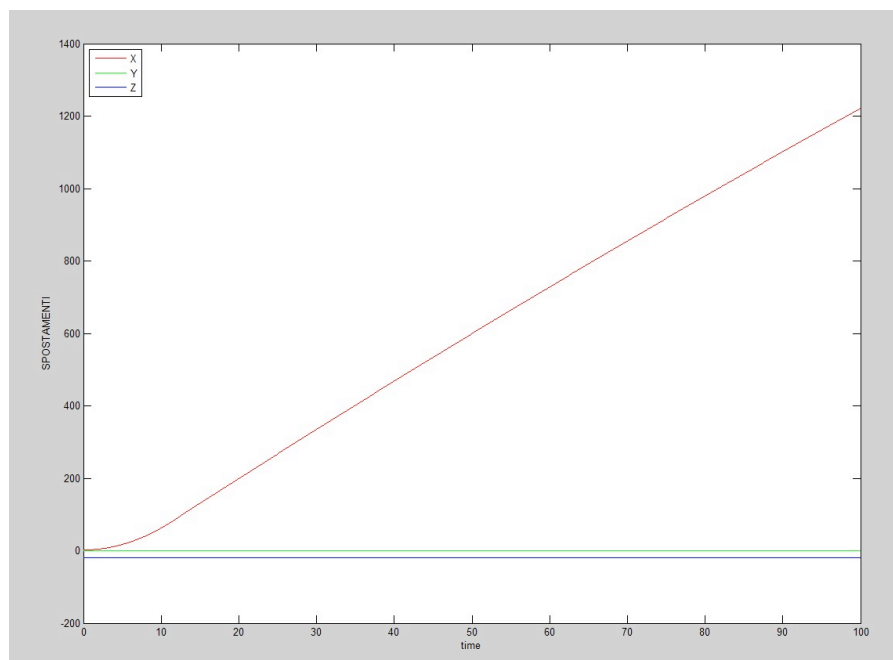


Fig.50: La parabola e la retta successiva in x.

Infine, si analizza il carico:

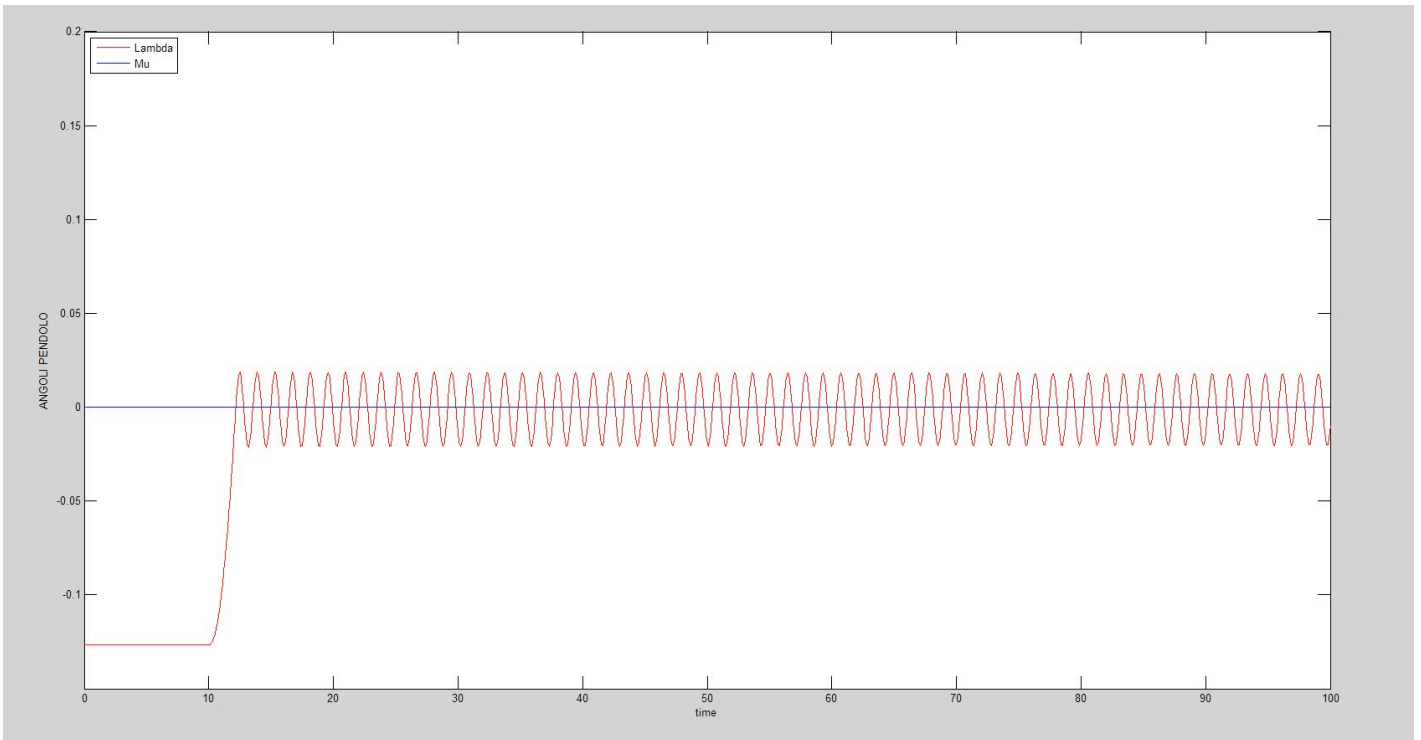


Fig.51: Il movimento del carico in questa simulazione.

dove si evince la stabilità del carico per un valore di $\lambda = 0$ e l'andamento successivo non smorzato dovuto alla cabrata successiva.

4.2.5 Manovra in Imbardata

L'ultima movimentazione riguarda quella eseguita nel terzo asse di rotazione del velivolo: la rotazione provoca una variazione dell'angolo di imbardata, ψ .

La condizione di partenza resta l'hover ad una quota data sempre di 40m e in seguito si ruoterà il velivolo prima a destra, successivamente a sinistra ed infine tornerà all'assetto iniziale.

Innanzitutto l'assetto che assumerà il drone nella simulazione:

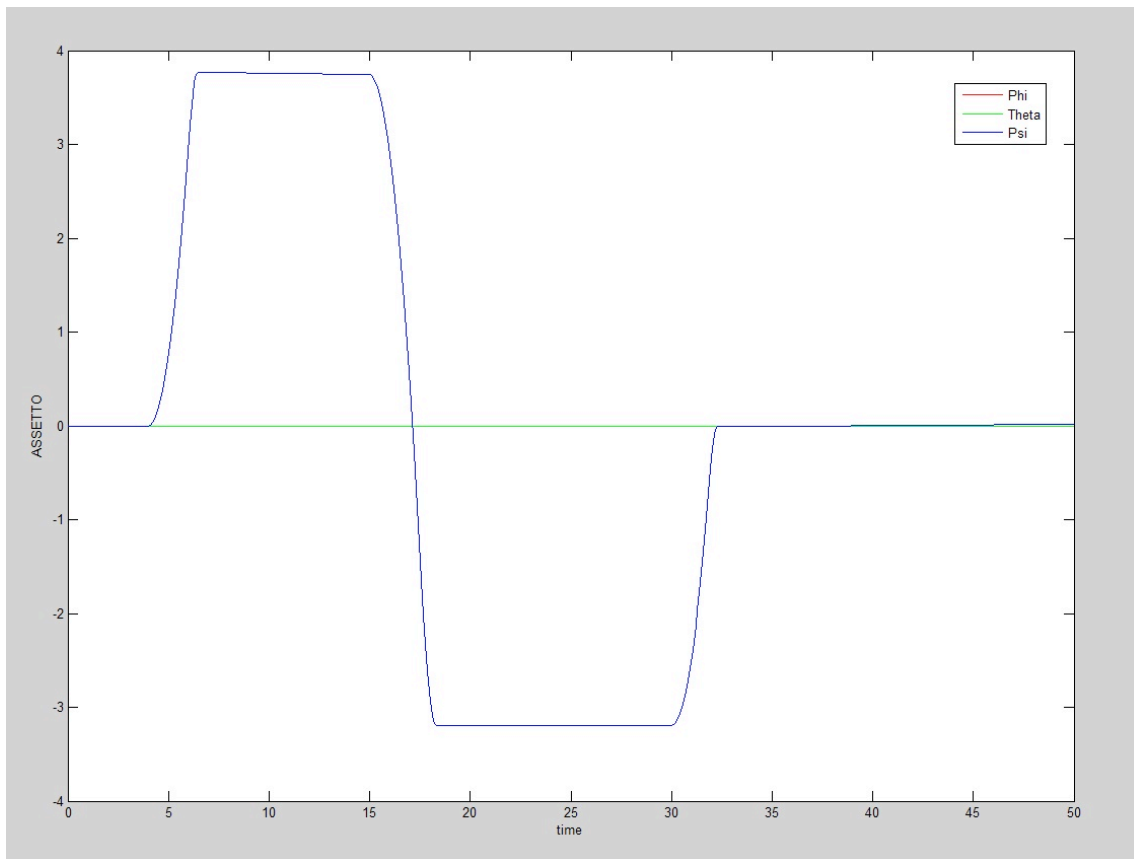


Fig.52: La variazione di ψ .

Per il funzionamento si sono dovuti pilotare anche qui i motori in maniera corretta in modo da realizzare una manovra comprensibilmente reale.

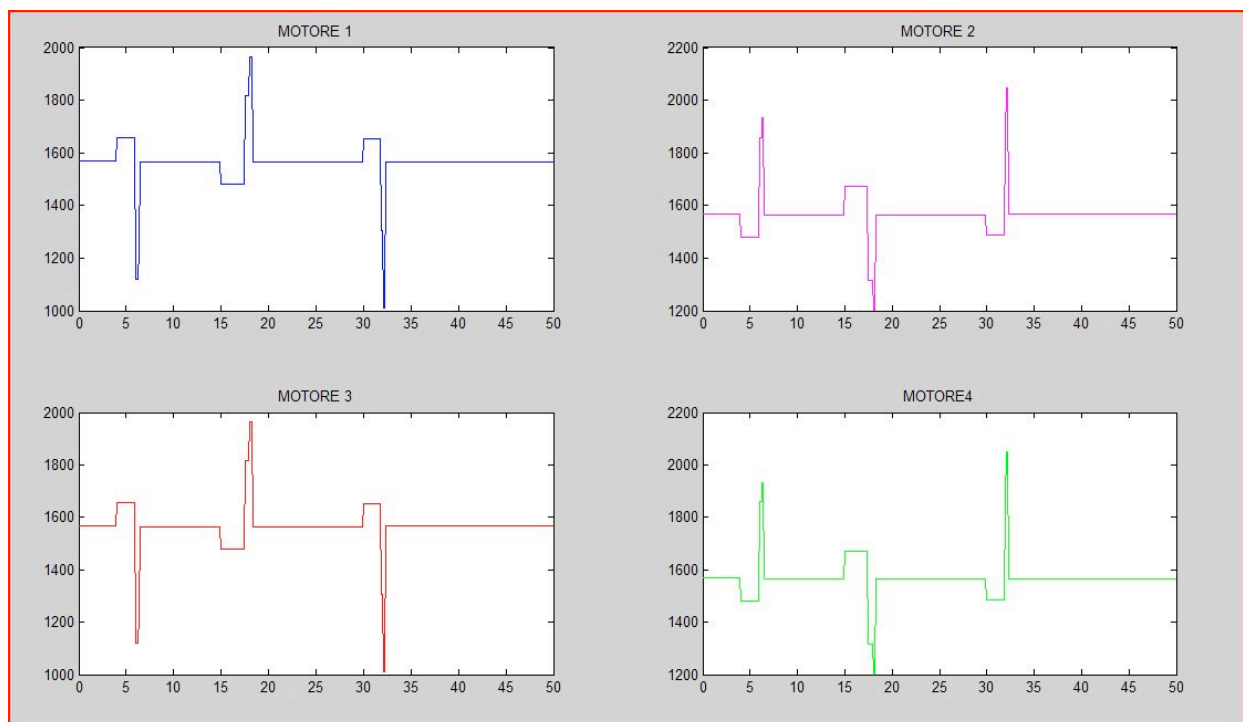


Fig.53: Le azioni opposte delle due coppie di rotori.

Come immaginato la coppia di motori 1-3 e 2-4 compiono variazioni uguali ed opposte in modo tale da preservare la quota il più possibile. In più le differenze di giri provocano lo scompenso di coppia necessario per effettuare l'imbardata. Per quanto riguarda il pendolo, esso da fermo all'istante iniziale resta tale poichè il velivolo pur muovendosi resta sul piano perpendicolare alla retta passante per il cavo. Quindi è corretto aver visualizzato:

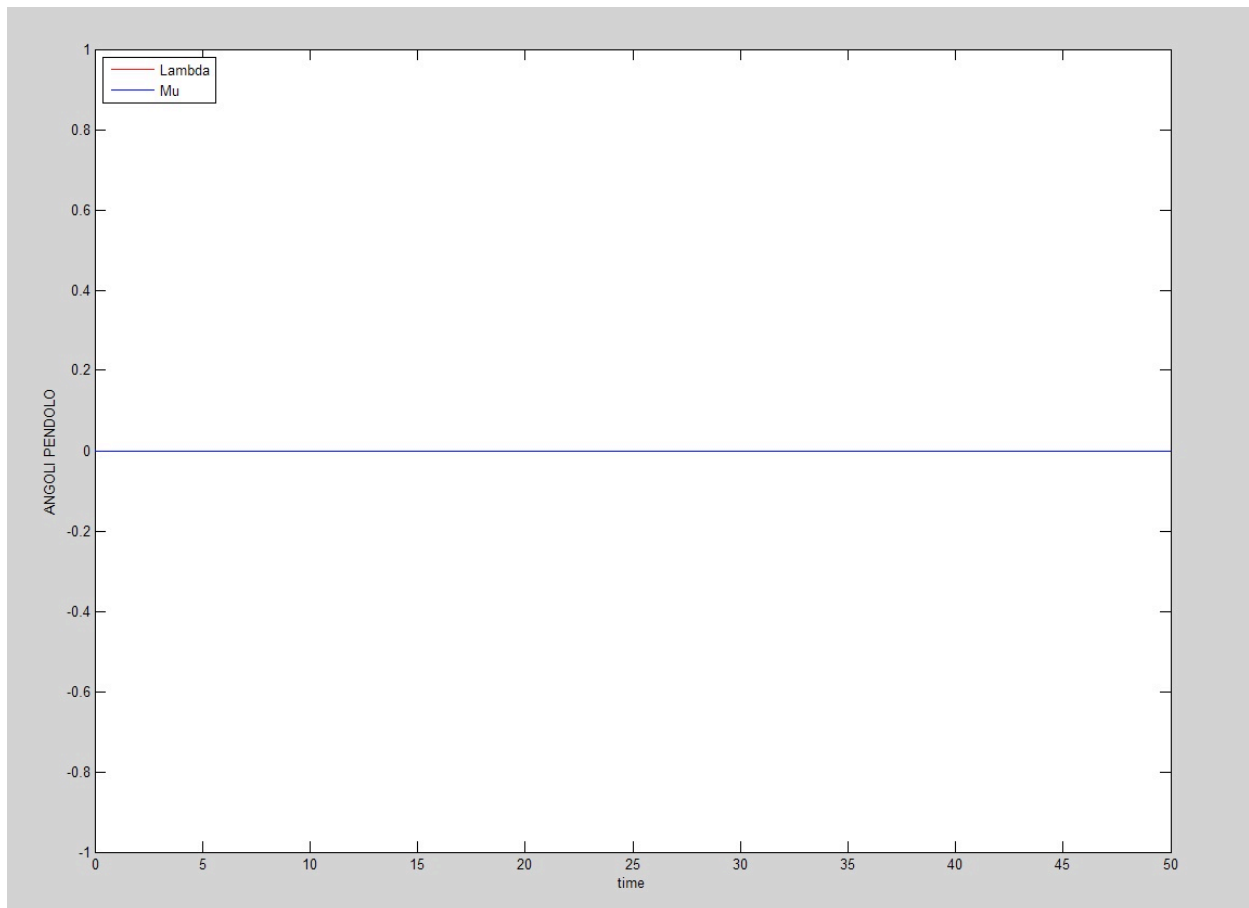


Fig.54:Il pendolo immobile in imbardata.

In una manovra di questo tipo ci si aspetta che il velivolo resti a quota costante e non compia quindi alcun movimento. Questo è vero nella direzione frontale e laterale del moto, x e y.

Per la direzione verticale z ciò non è perfettamente verificato: resta visibile l'attitudine a restare ad una data quota tuttavia l'imperfezione del segnale a gradino che assume valori discontinui negli istanti temporali considerati provoca delle variazioni. L'effetto quadratico del legame spinta/regimi motori provoca nei cambi di manetta in questo caso una variazione di altitudine.

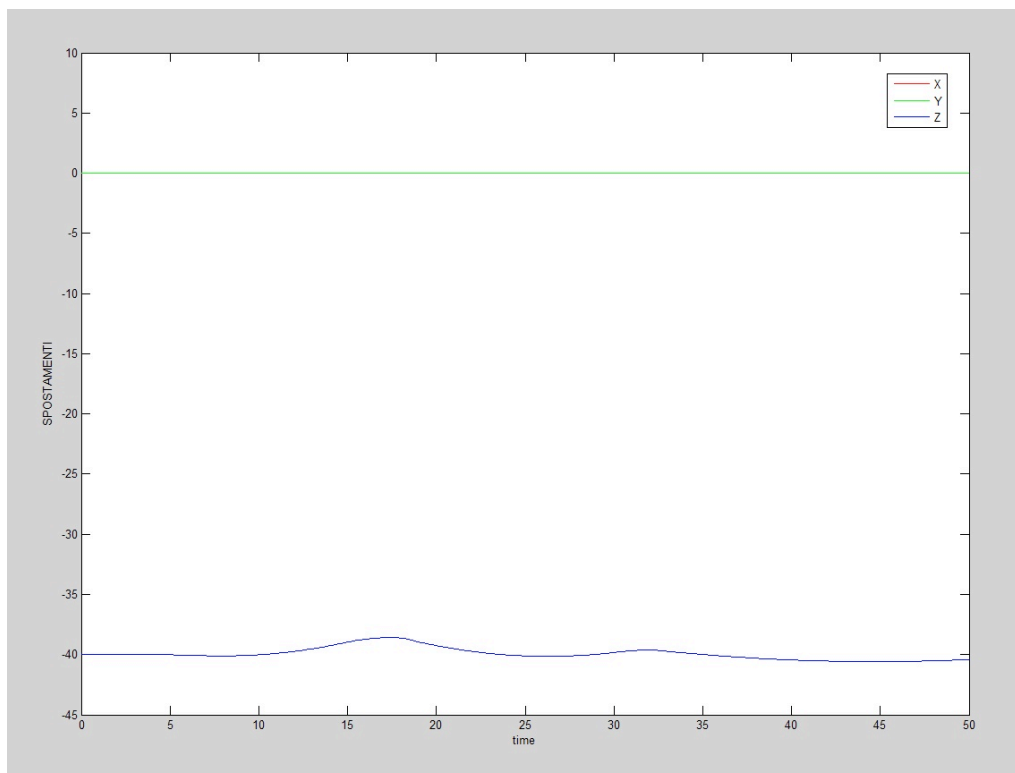


Fig.55: L'andamento globalmente costante delle posizioni.

Se si fosse invece cercato di mantenere la quota costante, non si sarebbe trovata una manovra pulita e definita come quella implementata.

Il fenomeno può essere visto anche nella variazione nell'andamento di tau, dove si nota infatti come il rapporto tra spinta e peso cambi e questo perciò si ripercuote anche sull'altitudine.

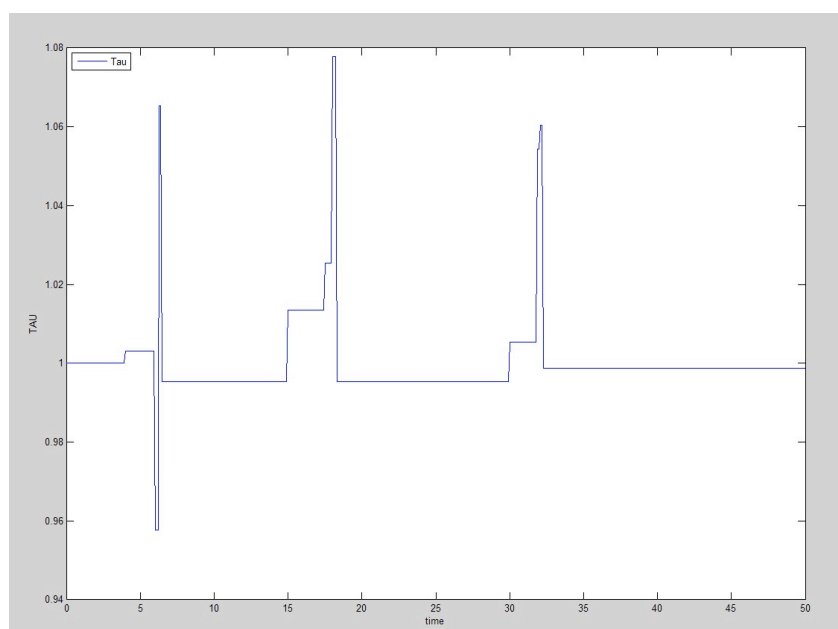


Fig.56: Tau in imbardata.

Conclusioni e Sviluppi futuri:

Nell'attività svolta si è sviluppato un metodo di comprensione e di calcolo in grado di quantificare le grandezze relative a spostamenti ed assetto di un quadricottero anche in presenza di payload attraverso lo studio delle equazioni del moto. In particolare si evince che il payload è una massa in più da considerare nel sistema capace di perturbare l'andamento delle grandezze misurate in funzione della sua oscillazione ma non cambiarle nel loro valor medio.

I modelli linearizzati sono risultati idonei nei limiti delle ipotesi imposte a monte delle modellazioni delle manovre di hovering, salita, traslazione frontale nella prima parte e di rotazione intorno agli assi locali del mezzo nella seconda. Essendo il lavoro presentato una breve parentesi di primo acchitto nel mondo dei multirotori, è necessario implementare step successivi per incrementare il realismo della simulazione aggiungendo opportunamente il fattore resistenza in primis nei due studi presentati. Può essere presa in considerazione ad esempio quella relativa ai quattro rotori o quella dello chassis ma anche, e specialmente, quella del payload soprattutto se il mezzo di propagazione di quest'ultimo non è l'aria.

Nel qual caso dovranno essere aggiunti così dei nuovi elementi rispettivamente nella formulazione di forze esterne in Eulero-Newton o nella formulazione di lavoro in Lagrange.

La stesura di un codice di maggior raffinatezza inoltre potrebbe essere arricchita con la progettazione di un sistema di controllo lineare di tipo PID per risposte a disturbi esterni quali raffiche di vento. Una diversa opportunità inoltre è creare un collegamento efficace tra interfaccia CAD e Matlab tramite strumenti come SimMechanics così da vedere gli effetti sul corpo stesso in tempo reale per una migliore e dettagliata verifica.

Bibliografia:

- Structural Design and Non-Linear Modeling of a Highly Stable Multi-Rotor Hovercraft, *Ali Shahbaz Haider, Muhammad Sajjad*;
- Kinematic Analysis and Control Design for a Nonplanar Multicopter Vehicle, *Bill Crowter, Alexander Lanson, Martin Maya-Gonzalez, David Langkamp*;
- Modeling and Simulation of a Quad-Tilt Rotor Aircraft, *Fabrizio Giuliatti, Gastone Ferrarese*;
- Transporting Hanging Loads Using a Scale Quad-Rotor, *Cesareo Raimundez, Luis Camacho*;
- Accurate Positioning of a Payload Suspended to a Quadrotor Unmanned Aircraft, *Giulio Avanzini, Fabrizio Giuliatti, Guido De Matteis*.

Sitografia:

Fig.1: <http://www.chiefaircraft.com/dji-fw550.html>

Fig.2: <http://rtsys.informatik.uni-kiel.de/confluence/display/RTProjectSSI5/Microcontroller>

Fig.4,5,6,7: http://wpage.unina.it/framato/materiale%20didattico/Tesine/quadricottero/Tesina_Quadricottero.pdf

Fig.8: https://it.wikipedia.org/wiki/Angoli_di_Eulero

Ringraziamenti:

Prima di tutti, vorrei ringraziare il Prof. Fabrizio Giulietti e l'Ing. Gastone Ferraresi, per la pazienza e la disponibilità messami a disposizione in questi mesi, per avermi messo in contatto e dato l'occasione di introdurmi in questo ambito per me nuovo.

Non di seconda importanza, un sincero grazie alla mia famiglia che ha da sempre posto un'incrollabile fiducia nei miei confronti e che, spronandomi, ha rappresentato un costante punto di riferimento in questi anni; sperando di avervi reso orgogliosi perché se oggi ho la possibilità di vivere questo giorno è soprattutto merito vostro.

Desidero inoltre ringraziare il Prof. Ceruti per la cortesia che mi ha mostrato in molte situazioni nel corso della mia carriera universitaria.

Non possono inoltre mancare loro, quelli che da compagni di studi sono diventati una parte essenziale della mia permanenza in questa città: il mio coinquilino e conterraneo Davide Grignoli, e gli insostituibili Lorenzo Lazzarini, Lorenzo Paolucci e Pietro Stuppa, con cui ho condiviso tensioni e svaghi di ogni tipo.

Dulcis in fundo, un grazie anche a me che, forte del fortunato ambiente di cui mi sono circondato, sono giunto ad un primo traguardo personale così importante.